



Proyecto de Sistemas Informáticos

Curso 2013/2014

UCMoocs

Una plataforma de Moocs de código abierto y orientada a universidades

Profesor director: Antonio Sarasa Cabezuelo

Autores: Alejandro Armenteros Rodelgo, Cristian García García, José Félix Martín Ramírez



Índice

Título	Página
• Índice de figuras	1
• Índice de abreviaturas	2
• Resumen	3
• Abstract	4
• Capítulo 1: Introducción	5
• Capítulo 2: Objetivos	8
• Capítulo 3: Tecnologías	10
• Capítulo 4: Especificación de casos de uso	15
▪ Casos de uso: Usuario general.....	16
▪ Casos de uso: Usuario registrado	19
▪ Casos de uso: Usuario actuando como alumno	22
▪ Casos de uso: Usuario actuando como profesor.....	27
▪ Casos de uso: Foro.....	38
• Capítulo 5: Arquitectura del sistema.....	42
▪ Arquitectura general del sistema	43
▪ Arquitectura de la base de datos	45
▪ Arquitectura de la aplicación.....	47
• Capítulo 6: Trabajo futuro	50
• Capítulo 7: Conclusiones	52
• Bibliografía.....	54
• Autorización a la UCM	55
• Anexos	56



Índice de figuras

Título	Página
• Figura 1: Diagrama de la arquitectura general de la plataforma	41
• Figura 2: Diagrama Entidad-Relación de la base de datos	43
• Figura 3: Diagrama general del patrón MVC	45



Índice de abreviaturas

- **MOOC:** Massive Online Open Course – Curso online masivo y abierto. Es una educación abierta que ofrece cursos gratuitos a un público amplio a través de Internet.
- **MVC:** Modelo-Vista-Controlador. Es un patrón de diseño de la arquitectura de un sistema informático que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario.
- **RoR:** Ruby on Rails. Es un framework de aplicaciones web escrito en el lenguaje de programación Ruby para desarrollar aplicaciones según el patrón MVC.
- **JSON:** JavaScript Object Notation. Es un formato ligero orientado al intercambio de datos.
- **BBDD:** Siglas correspondientes a Base de Datos.
- **HTML:** Siglas de HyperText Markup Language. Es el lenguaje de marcado utilizado para representar páginas web.
- **CSS:** Cascading Style Sheets – Hojas de estilo en cascada. Es un lenguaje de hojas de estilo utilizado para describir el aspecto y el formato de un documento html.



Resumen

UCMoocs. Una plataforma de Moocs de código abierto y orientada a universidades.

El proyecto que a continuación se describe consiste en una plataforma de MOOC (Massive Online Open Course) orientada al uso de los mismos en instituciones universitarias y educativas en general.

Una plataforma de MOOC es una modalidad de educación abierta que consiste en cursos ofrecidos gratuitamente a través de plataformas educativas en Internet cuya filosofía es la liberación del conocimiento para que este llegue a un público más amplio. De esta manera, la plataforma desarrollada ofrecerá cursos online que cualquiera puede realizar.

Además, la plataforma ofrece la posibilidad a cualquier usuario registrado para que pueda compartir cursos con el resto de la gente. De esta manera, cualquier usuario perteneciente a la plataforma puede actuar con dos roles: el de profesor y el de alumno.

Los cursos están basados en recursos, los cuales ofrecen diversos modos de presentar la información que se quiere poner a disposición de los alumnos para su aprendizaje. Estos recursos están agrupados en lecciones, las cuales representan un tema concreto de la materia total del curso. A su vez, estas lecciones están agrupadas en módulos, el cual es el nivel más alto de agrupación de temáticas.

De esta manera un alumno, para poder completar el curso, ha de realizar los distintos módulos del mismo y aprobar un examen final. En el caso del profesor, éste ha de crear recursos para poder crear lecciones que agrupar en módulos y así ofrecer toda la materia del curso a los alumnos.

La plataforma ha sido desarrollada en Ruby on Rails utilizando el patrón Modelo-Vista-Controlador (MVC). Se ha utilizado con la base de datos MongoDB, un sistema de bases de datos NoSQL basado en documentos JSON.

A continuación se explican los motivos que han motivado al grupo a desarrollar esta plataforma, así como la metodología y la manera mediante la cual se ha desarrollado.

Palabras Clave: curso, masivo, internet, educación, alumno, profesor, ruby, mongo, lección.



Abstract

UCMoocs. An open source Moocs platform for universities.

The project explained in this document describes a MOOC (Massive Online Open Course) platform oriented to university institutions and, in general, to educational institutions.

A MOOC platform is an open education method which consists in some free courses offered through education online platforms whose purpose is the knowledge exoneration in order to reach a broad range of audience. In this way, the developed platform will offer online course which can be completed by anyone.

In addition, the platform offers to any registered user the possibility to share his own courses with the rest of the people. With this possibility, any user of the platform can work with two roles: teacher and student.

Courses are based in resources, which offers diverse ways to display the information required to be available for the student's learning. These resources are grouped in lessons, which represent a specific topic of the course's complete subject. In addition, these lessons are grouped in lectures, which are the highest thematic aggrupation level.

The student, in order to complete the course, may complete all the lectures and pass the final test. In relation to the teacher role, he must create resources to be able to create lessons and group them into lectures in order to offer all the topics to the students.

The platform has been developed on Ruby on Rails using the Model-View-Controller (MVC) pattern. As data base system, MongoDB has been used. This system consists in a NoSQL database based in JSON documents.

Below in this text the reasons which have motivated the group to develop this platform are explained, as well as the methodology used.

Key Words: course, massive, internet, education, student, teacher, ruby, mongo, lesson



Capítulo 1:

Introducción



Internet está creciendo exponencialmente a lo largo de los últimos años. Se conocen múltiples utilidades que esta tecnología puede ofrecer como pueden ser: búsqueda de información, envío de mensajería, entretenimiento audiovisual, comercio electrónico, etc.

Además, la red es una de las mejores maneras de llegar a multitud de gente y compartir con la misma cualquier cosa que se nos pueda ocurrir. Este aspecto se puede utilizar para transmitir información a la gente y que todo el mundo la conozca y tenga acceso a ella.

Con esta idea llegamos al concepto de “aprendizaje y educación en la web”. La gente puede buscar información sobre una materia y recopilarla para conocer aquello que estaban buscando, aunque muchas veces esta información se encuentra dispersa por distintas páginas web y en distintos formatos e idiomas.

Por tanto parece lógico pensar en la utilidad que supone recopilar toda la información sobre una materia concreta, juntarla en un mismo sitio web y compartirla con todo el mundo. En este concepto se basan los Massive Online Open Courses (MOOC), que traducido al español significa “cursos en línea masivos y abiertos”.

Un MOOC es una modalidad de educación abierta que consiste en cursos ofrecidos gratuitamente a través de plataformas educativas en Internet cuya filosofía es la liberación del conocimiento para que este llegue a un público más amplio. Este término fue acuñado en el año 2008 por Dave Cormier cuando el número de inscritos a uno de sus cursos aumentó hasta casi 2300 estudiantes.

Para que la enseñanza a distancia pueda ser considerada MOOC debe cumplir los siguientes requisitos:

- Ser un curso: Debe contar con una estructura orientada al aprendizaje, que suele conllevar una serie de pruebas o evaluaciones para acreditar el conocimiento adquirido.
- Tener carácter masivo: El número de posibles matriculados es, en principio, ilimitado; o bien en una cantidad muy superior a la que podría contarse en un curso presencial. El alcance es global.
- En línea: El curso es a distancia e Internet es el principal medio de comunicación. No requiere la asistencia a un aula.
- Abierto: Los materiales son accesibles de forma gratuita en Internet. Ello no implica que puedan ser reutilizados en otros cursos.



La estructura general de un MOOC consiste en:

- Varios temas sobre una materia, generalmente con un examen al final de cada uno de los mismos.
- Cada tema se encuentra dividido en lecciones, las cuales se orientarán a fijar una serie de conceptos concretos. Al final de cada lección se pueden requerir ejercicios y test para garantizar la fijación de conceptos.
- Al finalizar todo el temario, se suele realizar un examen final que contiene preguntas o ejercicios referidos a cada uno de los temas. Con este examen se consigue conocer si el alumno ha entendido y aprendido los conceptos generales de la materia.

Las lecciones son difundidas mediante distintos recursos. Los principales tipos de recursos utilizados son: textos, audios, vídeos y videoconferencias. Estos recursos son añadidos a la plataforma Web por alguno de los profesores del curso y pueden ser consultados por todos los alumnos suscritos al mismo. Los profesores del curso también se encargan de dividir el temario en lecciones y proveer a las mismas de los recursos necesarios para su aprendizaje, así como de preparar los ejercicios, test y exámenes.

Desde la acuñación del término en 2008 hasta hoy, la cantidad de MOOC disponibles en la red ha crecido de manera vertiginosa, habiendo sido creadas diversas plataformas que ofrecen las facilidades necesarias para crear y difundir cursos de estas características.

El proyecto que se expone a continuación describe una plataforma para MOOC, con características innovadoras que no ofrecen aquellas actualmente disponibles en nuestro círculo accesible o nuestro idioma.

Para los autores de este trabajo las características referidas se hacen totalmente necesarias en una plataforma de MOOC, lo que despierta el interés y la necesidad de realizar una plataforma que incorpore dichas características.



Capítulo 2:

Objetivos



El objetivo principal del proyecto que se expone en la presente memoria es el diseño, implementación e implantación de una plataforma de MOOC libre, abierta y accesible a todo el mundo.

Para conseguir estas características, se proponen los siguientes objetivos a conseguir:

- Desarrollar una plataforma en software de código abierto (Open Source Software, OSS) que cualquier persona pueda utilizar y modificar según sus conveniencias.
- Orientar la plataforma a la estructura de los MOOC, es decir, hacer que soporte la división de información y catalogación de la misma en temas y lecciones, el alojamiento de métodos audiovisuales y documentales, la creación de ejercicios y exámenes, etc.
- Conseguir un sistema de almacenamiento lo suficientemente grande y escalable como para contener múltiples cursos con sus correspondientes contenidos; así como usuarios registrados, mensajes enviados entre usuarios, comentarios y sugerencias, etc.
- El sistema ha de ser de fácil uso y configuración, para que cualquiera que desee compartir información pueda hacerlo y crear un MOOC de una manera rápida y efectiva.
- Para conseguir llegar a la mayor cantidad posible de gente, la plataforma ha de soportar multi-idioma.
- El servidor desde el que se ejecute la plataforma ha de soportar una gran cantidad de usuarios simultáneos o escalar en función de su número.



Capítulo 3:

Tecnologías



Para el desarrollo e implantación de la plataforma se han utilizado diversas tecnologías, todas ellas con licencia de software libre para conseguir que nuestro proyecto también tenga este tipo de licencia.

Para el desarrollo del código se ha utilizado:

- **Ruby:** Es un lenguaje de programación interpretado, reflexivo y orientado a objetos. Es interpretado en una sola pasada y su implementación oficial es distribuida bajo una licencia de software libre. Está diseñado para la productividad y diversión del desarrollador, siguiendo los principios de una buena interfaz de usuario. También está considerado un lenguaje de programación multiparadigma: permite programación procedural, orientación a objetos y programación funcional.
- **RubyGems:** Es un gestor de paquetes para el lenguaje de programación Ruby que proporciona un formato estándar y autocontenido para poder distribuir programas o bibliotecas en Ruby. Incluye una herramienta destinada a gestionar la instalación de los paquetes y un servidor para la distribución de los mismos.
- **Ruby on Rails (RoR):** Es un framework de aplicaciones web de código abierto escrito en Ruby, siguiendo el paradigma de la arquitectura Modelo Vista Controlador (MVC). Su finalidad es combinar la simplicidad con la posibilidad de desarrollar aplicaciones escribiendo menos código que con otros frameworks y con un mínimo de configuración. Es distribuido a través de RubyGems.

Los principios fundamentales de Ruby on Rails incluyen:

- No te repitas (Don't repeat yourself, DRY): Significa que las definiciones deberían hacerse una sola vez. Los componentes están integrados de manera que no hace falta establecer puentes entre ellos.
 - Convención sobre configuración: Significa que el programador sólo necesita definir aquella configuración que no es convencional.
- **HTML:** Son las siglas de HyperText Markup Language (lenguaje de marcas interpretado) y hacen referencia al lenguaje de marcado utilizado para la elaboración de páginas web. Es un estándar a cargo de la W3C y con él se definen las páginas web. El lenguaje HTML basa su filosofía de desarrollo en la referenciación, es decir, para añadir un elemento externo a la página se hace una



referencia a la ubicación del mismo mediante texto. De este modo, la página web contiene sólo texto mientras que recae en el navegador web la tarea de unir todos los elementos y mostrar la página final.

- **CSS:** Son las siglas de Cascading Style Sheets (hoja de estilo en cascada). Es un lenguaje de hojas de estilo utilizado para describir el aspecto y el formato de un documento escrito en HTML. Cada hoja de estilos es una lista de reglas que contienen los estilos a aplicar a los elementos de la página web.
- **JavaScript:** Es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y la creación de páginas web dinámicas. Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web.
- **Bootstrap:** Es un framework de software libre orientado al diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografías, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basados en HTML y CSS, así como extensiones de JavaScript adicionales.

De esta manera, se ha utilizado Ruby on Rails para la implementación de la aplicación web junto con HTML, CSS y Bootstrap para la interfaz gráfica y el diseño de las páginas web.

Para la base de datos se ha optado por utilizar el sistema MongoDB (de la palabra “humongous” que significa enorme). MongoDB es un sistema de base de datos NoSQL, desarrollado por 10gen, orientado a documentos y desarrollado bajo el concepto de código abierto.

MongoDB forma parte de la nueva familia de sistemas de bases de datos NoSQL. En vez de utilizar tablas, como las bases de datos relacionales, MongoDB guarda estructuras de datos en documentos tipo JSON con un esquema dinámico (BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.



Las características principales de MongoDB son:

- **Consultas Ad hoc:** MongoDB soporta la búsqueda por campos, consulta de rangos y expresiones regulares.
- **Indexación:** cualquier campo de un documento puede ser indexado, también es posible hacer índices secundarios.
- **Replicación:** MongoDB soporta el tipo de replicación maestro-esclavo. El maestro puede ejecutar comandos de lectura y escritura, mientras que el esclavo puede copiar los datos del maestro y sólo se puede usar para lectura o para copia de seguridad, pero no se pueden realizar escrituras.
- **Balanceo de carga:** MongoDB se puede escalar de forma horizontal usando el concepto de “database shard”. “Sharding” es el proceso de almacenar registros de datos en múltiples máquinas para hacer frente a la demanda de almacenamiento según el proceso de crecimiento de los datos.
- **Almacenamiento de archivos:** MongoDB puede ser utilizado como un sistema de archivos, tomando la ventaja que ofrece en cuanto al balanceo de carga y la replicación de datos. Esto se puede realizar mediante la especificación GridFS ofrecida por MongoDB.
- **Agregación:** con la función MapReduce ofrecida por MongoDB se puede conseguir un procesamiento por lotes de datos y operaciones de agregación.

MongoDB se integró con Ruby mediante una RubyGem que hay disponible para ello.

Como sistema de control de revisión y desarrollo de software se ha escogido GitHub. Es una plataforma para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública y, para cada proyecto, se puede crear una wiki, una página web, gráficos para las colaboraciones de los desarrolladores y las bifurcaciones del proyecto, etc.

El código de la aplicación se encuentra alojado en la siguiente dirección de GitHub:

<https://github.com/felixmartin/moocs.git>



Para la implantación y pruebas de la aplicación se ha utilizado Heroku, una plataforma como servicio (Platform as a Service, PaaS) que soporta varios lenguajes de programación. Es una de las primeras plataformas de computación en la nube que apareció. La aplicación UCMoocs se encuentra alojada en la siguiente dirección de Heroku: <http://moocs.herokuapp.com/>

Además, como plataforma de desarrollo colaborativa se ha utilizado Nitrous.io. Se trata de una plataforma como servicio que ofrece máquinas virtuales en las que ejecutar la plataforma desarrollada, además de ofrecer un entorno de desarrollo online para poder modificar el código y ver los cambios en tiempo real en el servidor local.

Para la coordinación entre los miembros del equipo se ha utilizado producteev (www.producteev.com), una plataforma gratuita que permite la gestión y el seguimiento de tareas entre los miembros del equipo.



Capítulo 4:

Especificación de casos de uso



Casos de uso: Usuario general

El sistema está basado en un sistema de registro y login, es decir, para poder utilizar el sistema en su totalidad, es necesario registrarse para crear un usuario.

Si un usuario no está registrado, únicamente podrá realizar las siguientes acciones:

- Ver los cursos activos.
- Mostrar información de un curso.
- Registrarse.
- Buscar un curso.
- Reportar feedback.

Estas acciones están documentadas en las siguientes fichas de casos de uso:

ID Caso de Uso:	CU-01-GEN
Nombre:	Ver cursos activos
Objetivo:	Mostrar una lista con los cursos actualmente activos en el sistema.
Actores:	Cualquier usuario que acceda al sistema
Precondiciones:	Ninguna.
Postcondiciones:	La lista con los cursos activos ha de mostrarse por pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón "Cursos".2. Se consulta a la BBDD los cursos activos.3. Se muestra por pantalla una nueva página web con la lista de cursos disponibles en el sistema.
Escenarios alternativos:	<ol style="list-style-type: none">1. Si la lista de cursos es vacía, se muestra un mensaje con la información.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-02-GEN
Nombre:	Mostrar información de un curso
Objetivo:	Mostrar la información de un curso.
Actores:	Cualquier usuario que acceda al sistema.
Precondiciones:	CU-01-GEN / CU-04-GEN
Postcondiciones:	La información de un curso ha de mostrarse por pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el título de un curso.2. Se muestra por pantalla la información relativa al curso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-03-GEN
Nombre:	Registrarse
Objetivo:	Registrar a un nuevo usuario en el sistema y añadir su información en la BBDD.
Actores:	Usuario que accede al sistema y quiere registrarse.
Precondiciones:	El e-mail introducido no debe haber sido registrado previamente.
Postcondiciones:	Un nuevo usuario es añadido al sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón "Regístrate".2. Se muestra una página web con el formulario de registro.3. El usuario introduce sus datos y envía el formulario.
Escenarios alternativos:	<ol style="list-style-type: none">1. E-mail repetido:<ol style="list-style-type: none">a. Se notifica al usuario de que el e-mail introducido ya se encuentra registrado.b. Se muestra el formulario de registro con el e-mail resaltado en rojo.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-04-GEN
Nombre:	Buscar Curso
Objetivo:	Buscar un curso mediante palabras clave.
Actores:	Cualquier usuario que accede al sistema.
Precondiciones:	Encontrarse en la página principal de la plataforma.
Postcondiciones:	Una lista con los resultados de la búsqueda ha de ser mostrada por pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario introduce las palabras a buscar en la barra de búsqueda.2. El usuario pulsa el botón “Buscar”.3. El sistema consulta la BBDD.4. Se muestran por pantalla la lista con los cursos encontrados.
Escenarios alternativos:	<ol style="list-style-type: none">1. No se encuentra ningún curso:<ol style="list-style-type: none">a. Se notifica al usuario de que no se ha encontrado ningún curso.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-05-GEN
Nombre:	Reportar Feedback
Objetivo:	Reportar un formulario de feedback a los administradores de la plataforma.
Actores:	Cualquier usuario que acceda a la plataforma.
Precondiciones:	Ninguna.
Postcondiciones:	Un mensaje de feedback ha sido enviado a los administradores.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón de “Feedback” situado a la izquierda de su pantalla.2. Se abre un pop-up con un formulario.3. El usuario rellena el formulario y pulsa “Enviar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



Casos de uso: Usuario registrado

Cualquier usuario registrado puede adoptar dos roles: alumno y profesor. De esta manera, un usuario puede seguir y realizar todos los cursos que le interesen y estén disponibles en ese momento, pero también puede crear sus propios cursos y ofrecerlos a los demás usuarios.

Un usuario registrado, en general, sin distinguir si ejerce como alumno o como profesor en ese momento, tiene disponibles las siguientes acciones en el sistema:

- Iniciar sesión.
- Cerrar sesión.
- Gestionar perfil.
- Editar perfil.

Estas acciones están documentadas en las siguientes fichas de casos de uso:

ID Caso de Uso:	CU-01-REG
Nombre:	Iniciar sesión
Objetivo:	Crear una nueva sesión para un usuario registrado.
Actores:	Un usuario que se ha registrado previamente.
Precondiciones:	Haber completado el registro en la aplicación.
Postcondiciones:	Haber creado una nueva sesión para el usuario registrado
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Entra”.2. Aparece una nueva página web con el formulario de Log In.3. El usuario introduce su e-mail y su contraseña y pulsa el botón “Log in”.
Escenarios alternativos:	<ol style="list-style-type: none">1. E-mail no registrado:<ol style="list-style-type: none">a. Se notifica al usuario de que el e-mail no se encuentra registrado.b. Se muestra la página principal de la aplicación.2. Password incorrecto:<ol style="list-style-type: none">a. Se notifica al usuario de que el password es incorrecto.b. Se muestra de nuevo el formulario de Log In.



Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.
----------------------------	--

ID Caso de Uso:	CU-02-REG
Nombre:	Cerrar sesión
Objetivo:	Cerrar una sesión existente.
Actores:	Un usuario que ha iniciado sesión previamente.
Precondiciones:	Haber iniciado sesión previamente.
Postcondiciones:	La sesión activa ha de ser cerrada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón "Salir".2. Se muestra la página principal de la aplicación con la sesión cerrada.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al cerrar la sesión:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-03-REG
Nombre:	Gestionar perfil
Objetivo:	Gestionar el perfil de un usuario identificado.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-01-REG
Postcondiciones:	La página de gestión de perfil ha de estar mostrada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón con su correo electrónico en la barra superior de la plataforma.2. Se muestra una nueva página web con el perfil del usuario, que contiene su información y los cursos vinculados.
Escenarios alternativos:	<ol style="list-style-type: none">1. La lista de cursos vinculados es vacía:<ol style="list-style-type: none">a. En lugar de la lista de cursos vinculados, se muestra un mensaje informando al usuario de la situación.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-04-REG
Nombre:	Editar perfil
Objetivo:	Editar el perfil de un usuario identificado.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-03-REG
Postcondiciones:	La página de edición de perfil ha de estar mostrada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón "Editar perfil"2. Se muestra una nueva página web con el formulario para editar el perfil del usuario.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



Casos de uso: Usuario actuando como alumno

Cuando un usuario registrado ejerce como rol de Alumno, este debe suscribirse a un curso para posteriormente desarrollar un número de lecciones determinado que superará si aprueba los test o autoevaluaciones que se encuentran al final de las mismas. Una vez superadas todas las lecciones, el alumno debe enfrentarse a un examen final para conseguir superar el curso en su totalidad.

De esta manera, un usuario con el rol de alumnos, tiene disponible las siguientes acciones:

- Suscribirse a un curso.
- Gestionar curso suscrito.
- Ver módulos de un curso.
- Ver lecciones de un módulo.
- Ver recursos de una lección.
- Utilizar recurso de una lección.
- Realizar test.
- Realizar examen final.
- Mostrar progreso

Estas acciones están documentadas en las siguientes fichas de casos de uso:

ID Caso de Uso:	CU-01-ALU
Nombre:	Suscribirse a un curso
Objetivo:	Suscribirse a un curso de los disponibles en el sistema.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	Haber iniciado una sesión previamente.
Postcondiciones:	El usuario ha sido suscrito al curso seleccionado.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario busca cursos o accede al menú de cursos.2. El usuario selecciona un curso.3. El usuario pulsa en el botón "Unirse" situado a la derecha de la pantalla.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.



	b. Se muestra la página de inicio de la aplicación.
--	---

ID Caso de Uso:	CU-02-ALU
Nombre:	Gestionar curso suscrito
Objetivo:	Acceder al menú de un curso suscrito.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-03-REG
Postcondiciones:	Se muestra por pantalla la información de un curso y el menú de gestión.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario selecciona el curso a gestionar en la lista de cursos de su perfil.2. Se muestra una nueva página web con la información y el menú del curso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-03-ALU
Nombre:	Ver módulos del curso
Objetivo:	Acceder a la página de los módulos del curso.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-02-ALU
Postcondiciones:	Se muestra por pantalla la lista de módulos del curso.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario selecciona "Módulos" en el menú de gestión del curso.2. Se muestra una nueva página web con la lista de los módulos del curso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-04-ALU
Nombre:	Ver lecciones de un módulo
Objetivo:	Acceder a la lista de lecciones de un módulo.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-03-ALU
Postcondiciones:	Se muestra por pantalla la lista de lecciones que componen un módulo.



Flujo de eventos:	<ol style="list-style-type: none">1. El usuario hace click en el título de un módulo de la lista de módulos.2. Se muestra una nueva página web con la lista de lecciones que componen dicho módulo.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-05-ALU
Nombre:	Ver recursos de una lección
Objetivo:	Acceder a la lista de recursos que componen una lección.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-04-ALU
Postcondiciones:	El usuario visualiza la lista de recursos que componen una lección.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario hace click en el título de una de las lecciones de la lista de lecciones.2. Se muestra una nueva página web con la lista de recursos que componen la lección.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-06-ALU
Nombre:	Utilizar recurso de una lección
Objetivo:	Utilizar un recurso de los que componen una lección.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-05-ALU
Postcondiciones:	El usuario ha utilizado el recurso en cuestión.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario hace click en el título de un recurso de los que aparecen en la lista de lecciones.2. El usuario ve el video, lee el documento o escucha el audio presente en el recurso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-07-ALU
Nombre:	Realizar test
Objetivo:	Realizar un test de un recurso o una lección.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-05-ALU o CU-06-ALU
Postcondiciones:	El usuario ha realizado un test
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario hace click en el recurso que se corresponde al test.2. El usuario contesta las preguntas según las indicaciones del test.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-08-ALU
Nombre:	Realizar examen final
Objetivo:	Realizar el examen final del curso.
Actores:	Un usuario con una sesión iniciada.
Precondiciones:	CU-04-ALU y haber completado todos los módulos del curso.
Postcondiciones:	El usuario ha realizado el examen final del curso.
Flujo de eventos:	<ol style="list-style-type: none">1. En el menú de gestión del curso, el usuario pulsa en el examen final.2. El usuario realiza el examen según las instrucciones mostradas en el mismo.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-09-ALU
Nombre:	Mostrar progreso
Objetivo:	Mostrar una gráfica con el progreso del curso.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	Se muestra una página web con la gráfica de progreso del curso.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Progreso” del menú de gestión del curso.2. Se muestra una página web con una gráfica que muestra el progreso del curso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



Casos de uso: Usuario actuando como profesor

Por otro lado, como rol de Profesor, este debe crear un curso con los módulos, las lecciones, los recursos y los exámenes que lo componen. Una vez creada la estructura del curso y el contenido del mismo, éste estará disponible para que otros usuarios se suscriban y lo realicen.

De esta manera un usuario, como rol de Profesor, tiene disponibles las siguientes acciones:

- Gestionar mis recursos.
- Crear nuevo recurso.
- Gestionar mis test.
- Crear nuevo test.
- Crear curso.
- Gestionar un curso creado.
- Desactivar curso.
- Reactivar curso.
- Crear aviso.
- Editar aviso.
- Eliminar aviso.
- Editar información del curso.
- Gestionar módulos.
- Crear módulo.
- Editar módulo.
- Eliminar módulo.
- Gestionar un módulo.
- Crear lección.
- Agregar recursos.
- Editar lección.
- Eliminar lección.
- Agregar test a un módulo.



Estas acciones están documentadas en las siguientes fichas de casos de uso:

ID Caso de Uso:	CU-01-PRO
Nombre:	Gestionar mis recursos
Objetivo:	Gestionar los recursos creados por un profesor.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	Haber iniciado una sesión previamente.
Postcondiciones:	La página de gestión de recursos ha de ser mostrada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón "Recursos".2. En el menú desplegable selecciona "Mis Recursos".3. Se muestra una nueva página web con los recursos del usuario.
Escenarios alternativos:	<ol style="list-style-type: none">1. La lista de recursos es vacía.<ol style="list-style-type: none">a. Se muestra un mensaje indicando que el usuario no ha creado ningún recurso.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-02-PRO
Nombre:	Crear nuevo recurso
Objetivo:	Crear un nuevo recurso a utilizar posteriormente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-01-PRO
Postcondiciones:	Un nuevo recurso, vinculado al usuario activo, ha sido creado en el sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón "Nuevo Recurso".2. Se muestra una nueva página web con un formulario a rellenar.3. El usuario rellena el formulario con los datos del recurso.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-03-PRO
Nombre:	Gestionar mis test
Objetivo:	Gestionar los test creados por un profesor.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	Haber iniciado una sesión previamente.
Postcondiciones:	La página de gestión de test ha de ser mostrada.



Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Recursos”.2. En el menú desplegable selecciona “Mis Test”.3. Se muestra una nueva página web con los test del usuario.
Escenarios alternativos:	<ol style="list-style-type: none">1. La lista de test es vacía.<ol style="list-style-type: none">a. Se muestra un mensaje indicando que el usuario no ha creado ningún test.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-04-PRO
Nombre:	Crear un test
Objetivo:	Crear un nuevo test en el sistema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-03-PRO
Postcondiciones:	Un nuevo test, vinculado al usuario activo, ha sido creado en el sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Nuevo Test”.2. Se muestra una nueva página web con un formulario a rellenar.3. El usuario rellena el formulario con los datos y las preguntas del test.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-05-PRO
Nombre:	Crear un curso
Objetivo:	Crear un nuevo curso en el sistema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	Haber iniciado una sesión previamente.
Postcondiciones:	Un nuevo curso, vinculado al usuario activo, ha sido creado en el sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Recursos”.2. Se muestra una lista desplegable y el usuario selecciona “Crear Curso”.3. Se muestra una página web con un formulario relativo a la información del curso4. El usuario rellena el formulario con los datos del curso.
Escenarios alternativos:	



Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.
----------------------------	---

ID Caso de Uso:	CU-06-PRO
Nombre:	Gestionar un curso creado
Objetivo:	Acceder a la página de gestión de un curso creado por el usuario.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-05-PRO – Haber creado un curso previamente. CU-03-REG – Acceder a la gestión del perfil.
Postcondiciones:	La página de gestión del curso es mostrada en la pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario selecciona un curso en el cual tenga el rol de profesor.2. Se muestra una página web con la información del curso y un menú de gestión del mismo.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-06-PRO
Nombre:	Gestionar un curso creado
Objetivo:	Acceder a la página de gestión de un curso creado por el usuario.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-05-PRO – Haber creado un curso previamente. CU-03-REG – Acceder a la gestión del perfil.
Postcondiciones:	La página de gestión del curso es mostrada en la pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario selecciona un curso en el cual tenga el rol de profesor.2. Se muestra una página web con la información del curso y un menú de gestión del mismo.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-07-PRO
Nombre:	Desactivar curso
Objetivo:	Desactivar un curso del sistema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	El curso gestionado es desactivado.
Flujo de eventos:	1. El usuario pulsa el botón “Desactivar” en el menú de gestión del curso.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-08-PRO
Nombre:	Reactivar curso
Objetivo:	Reactivar un curso del sistema que ha sido desactivado previamente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-07-PRO
Postcondiciones:	El curso gestionado es reactivado.
Flujo de eventos:	1. El usuario pulsa el botón “Activar” en el menú de gestión del curso.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-09-PRO
Nombre:	Crear aviso
Objetivo:	Crear un nuevo aviso vinculado al curso.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	Un nuevo aviso ha sido creado en el panel del curso.
Flujo de eventos:	1. El usuario pulsa el botón “Crear Aviso” en el menú de gestión del curso. 2. Se muestra una nueva página web con un formulario para informar de los datos del aviso. 3. El usuario rellena el formulario y pulsa el botón “Enviar”.
Escenarios alternativos:	



Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.
----------------------------	---

ID Caso de Uso:	CU-10-PRO
Nombre:	Editar aviso
Objetivo:	Editar un aviso creado anteriormente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-09-PRO
Postcondiciones:	El aviso ha sido editado.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Editar” de la parte superior derecha del aviso.2. Se muestra una nueva página web con un formulario para informar de los datos del aviso.3. El usuario modifica el formulario y pulsa el botón “Enviar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-11-PRO
Nombre:	Eliminar aviso
Objetivo:	Eliminar un aviso creado anteriormente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-09-PRO
Postcondiciones:	El aviso ha sido eliminado.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Eliminar” de la parte superior derecha del aviso.2. El usuario confirma la acción en el pop-up.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-12-PRO
Nombre:	Editar información del curso
Objetivo:	Editar la información correspondiente al curso seleccionado.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	La información vinculada al curso ha sido modificada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Editar” en el menú de gestión del curso.2. Se muestra una página web con un formulario que contiene los datos actuales del curso.3. El usuario modifica los datos pertinentes y pulsa en “Actualizar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-13-PRO
Nombre:	Gestionar módulos del curso
Objetivo:	Acceder al menú de gestión de los módulos del curso.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	El menú de gestión de módulos es mostrado.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Módulos” en el menú de gestión del curso.2. Se muestra una página web con la lista de módulos del curso.
Escenarios alternativos:	<ol style="list-style-type: none">1. La lista de módulos es vacía:<ol style="list-style-type: none">a. Se informa al usuario de que no existe ningún módulo para el curso.
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-14-PRO
Nombre:	Crear módulo
Objetivo:	Crear un nuevo módulo vinculado al curso actual.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-12-PRO
Postcondiciones:	Un nuevo módulo es creado en el sistema y vinculado al curso.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Nuevo Módulo” situado abajo a la derecha.2. Se muestra una página web con un formulario para



	introducir la información del módulo a crear. 3. El usuario rellena el formulario y pulsa el botón “Enviar”.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al introducir datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-15-PRO
Nombre:	Editar módulo
Objetivo:	Editar la información de un módulo creado previamente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-14-PRO
Postcondiciones:	La información correspondiente al módulo es modificada.
Flujo de eventos:	1. El usuario pulsa el botón “Actions” situado en la parte inferior del resumen de un módulo. 2. El usuario pulsa el botón “Editar”. 3. Se muestra un formulario con la información vinculada al módulo. 4. El usuario modifica la información pertinente y pulsa “Enviar”.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-16-PRO
Nombre:	Eliminar módulo
Objetivo:	Elimina un módulo creado previamente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-14-PRO
Postcondiciones:	El módulo ha sido eliminado del sistema.
Flujo de eventos:	1. El usuario pulsa el botón “Actions” situado en la parte inferior del resumen de un módulo. 2. El usuario pulsa el botón “Eliminar”. 3. El usuario acepta el pop-up de confirmación.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-17-PRO
Nombre:	Gestionar un módulo
Objetivo:	Acceder a la página de gestión de un módulo.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-14-PRO
Postcondiciones:	Se muestra la página de gestión del módulo.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el nombre del módulo a gestionar.2. Se muestra una página web con la información correspondiente al módulo seleccionado.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-18-PRO
Nombre:	Crear lección
Objetivo:	Crear una lección nueva en el sistema y vincularla al módulo que está siendo actualmente gestionado.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-17-PRO
Postcondiciones:	Una nueva lección ha sido creada en el sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Nueva Lección”.2. Se muestra una nueva página web con un formulario con los datos de la lección y un menú de gestión para la misma.3. En este momento se deben llevar a cabo los casos de uso CU-19-PRO y/o CU-02-PRO4. El usuario pulsa el botón “Enviar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-19-PRO
Nombre:	Agregar recursos
Objetivo:	Agregar recursos a una lección para que pueda ser creada
Actores:	Un usuario con sesión iniciada.
Precondiciones:	Encontrarse en el flujo de CU-18-PRO
Postcondiciones:	Un recurso es agregado a una lección en creación.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Agregar Recurso”.2. Aparece un pop up con los recursos que corresponden



	al usuario activo. 3. El usuario selecciona los recursos a agregar y pulsa el botón "Ok".
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-20-PRO
Nombre:	Editar lección
Objetivo:	Modificar los datos asociados a una lección.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-18-PRO
Postcondiciones:	La información asociada a una lección es modificada.
Flujo de eventos:	1. El usuario pulsa sobre el botón "Editar" de una lección. 2. Se muestra el formulario con la información asociada y los recursos asociados. 3. El usuario modifica la información del formulario o los recursos asociados mediante CU-19-PRO
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-21-PRO
Nombre:	Eliminar lección
Objetivo:	Eliminar una lección del sistema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-18-PRO
Postcondiciones:	La lección ha sido eliminada del sistema.
Flujo de eventos:	1. El usuario pulsa el botón "Eliminar" asociado a una lección. 2. El usuario confirma el pop-up que aparece.
Escenarios alternativos:	
Excepciones y tratamiento:	1. Error al modificar datos en la base de datos: a. Se notifica al usuario del error. b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-22-PRO
Nombre:	Agregar test a un módulo
Objetivo:	Agregar un test final a un módulo.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-17-PRO
Postcondiciones:	El test ha sido vinculado al módulo.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Seleccionar test” del menú de gestión del módulo.2. En el pop-up que se muestra, el usuario selecciona el test a añadir.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



Casos de uso: Foro

Por otro lado, hay funciones de comunicación entre alumnos y con el profesor. La principal comunicación horizontal entre usuarios y el profesor es el foro. Cada curso tiene un foro en el que, tanto el profesor creador del curso como los alumnos suscritos pueden publicar mediante la creación de temas y mensajes.

Esta herramienta contempla las siguientes acciones:

- Gestionar el foro.
- Crear tema.
- Editar tema.
- Eliminar tema.
- Mostrar los mensajes de un tema.
- Crear mensaje.
- Editar mensaje.
- Eliminar mensaje.

ID Caso de Uso:	CU-01-FOR
Nombre:	Gestionar el foro
Objetivo:	Mostrar el menú de gestión del foro asociado al curso.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-06-PRO
Postcondiciones:	El menú de gestión del foro aparece por pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón "Foro" del menú de gestión del curso.2. Se muestra una nueva página web con el menú de gestión del foro.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-02-FOR
Nombre:	Crear tema
Objetivo:	Crear un nuevo tema en el foro del curso.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-23-PRO



Postcondiciones:	Un nuevo tema ha sido creado en el foro del curso.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa el botón “Nuevo tema”.2. Se muestra por pantalla un formulario con los datos a rellenar del tema.3. El usuario introduce la información referente al nuevo tema y pulsa el botón “Crear tema”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al introducir datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-03-FOR
Nombre:	Editar tema
Objetivo:	Editar un tema del foro creado anteriormente.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-24-PRO
Postcondiciones:	La información asociada al tema ha sido modificada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Acciones” de un tema.2. El usuario pulsa el botón “Editar”.3. Se muestra un formulario con la información del tema.4. El usuario modifica la información y pulsa el botón “Modificar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-04-FOR
Nombre:	Eliminar tema
Objetivo:	Eliminar un tema del foro.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-24-PRO
Postcondiciones:	El tema ha sido eliminado del sistema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Acciones” de un tema.2. El usuario pulsa el botón “Eliminar”.3. Confirma el pop-up que aparece.



Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-05-FOR
Nombre:	Mostrar los mensajes de un tema
Objetivo:	Mostrar la página de mensajes de un tema del foro.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-24-PRO
Postcondiciones:	La página de mensajes del tema se muestra por pantalla.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el título de un tema.2. Se muestra una página web con los mensajes del tema.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al consultar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-06-FOR
Nombre:	Crear mensaje
Objetivo:	Crear un nuevo mensaje vinculado a un tema del foro.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-27-PRO
Postcondiciones:	Un nuevo mensaje vinculado al tema ha sido creado.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón "Crear Mensaje".2. Se muestra una página web con un formulario para rellenar los datos del mensaje.3. El usuario rellena el formulario y pulsa el botón "Crear Mensaje".
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al insertar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



ID Caso de Uso:	CU-07-FOR
Nombre:	Editar mensaje
Objetivo:	Editar un mensaje de un tema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-28-PRO
Postcondiciones:	La información asociada al mensaje ha sido modificada.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Acciones” de un mensaje.2. El usuario pulsa el botón “Editar”.3. Se muestra un formulario con la información del mensaje.4. El usuario modifica la información y pulsa el botón “Modificar”.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.

ID Caso de Uso:	CU-08-FOR
Nombre:	Eliminar mensaje
Objetivo:	Eliminar un mensaje de un tema.
Actores:	Un usuario con sesión iniciada.
Precondiciones:	CU-28-PRO
Postcondiciones:	El mensaje ha sido eliminado del tema.
Flujo de eventos:	<ol style="list-style-type: none">1. El usuario pulsa sobre el botón “Acciones” de un mensaje.2. El usuario pulsa el botón “Eliminar”.3. El usuario acepta el pop-up que se muestra.
Escenarios alternativos:	
Excepciones y tratamiento:	<ol style="list-style-type: none">1. Error al modificar datos en la base de datos:<ol style="list-style-type: none">a. Se notifica al usuario del error.b. Se muestra la página de inicio de la aplicación.



Capítulo 5:

Arquitectura del sistema



Arquitectura general del sistema

La arquitectura del sistema se basa en el patrón de diseño Modelo-Vista-Controlador (MVC), cuyo propósito y funcionamiento se explica con detalle en la sección correspondiente a la “Arquitectura de la aplicación”.

El modelo se apoya en la base de datos, donde se guardará toda la información referente a cada una de las clases y componentes del sistema.

Por otro lado, el servidor web se comunicará con los controladores gracias a la ayuda del enrutador (routes) y el distribuidor (dispatcher).

El flujo básico que se lleva a cabo con cada interacción del usuario con la página web es el siguiente:

1. El usuario realiza alguna acción en el navegador, como puede ser pulsar un botón, entrar en un enlace, etc.
2. El navegador envía una petición al servidor web señalando la acción que ha iniciado el usuario.
3. El servidor web procesa la petición, determina a que ruta pertenece la misma y la distribuye al método correspondiente del controlador.
4. El controlador pide al modelo la información que necesita para ser capaz de completar la petición.
5. El modelo recopila la información necesaria con el apoyo de la base de datos y se la facilita al controlador.
6. El controlador ofrece la información necesaria a la vista e insta a la misma a formarse con la información proporcionada.
7. La vista recoge la información y devuelve el código html formado al controlador.

8. El controlador junta la página devuelta por la vista con el código html de su propiedad y forma la página web completa que se mostrará al usuario. Después envía esta página al servidor web.
9. El servidor web devuelve la página al navegador para que éste la muestre al usuario.

Este ciclo se repite durante todo el uso de la aplicación y, de la manera descrita, el usuario interacciona con las páginas web para modificar los datos.

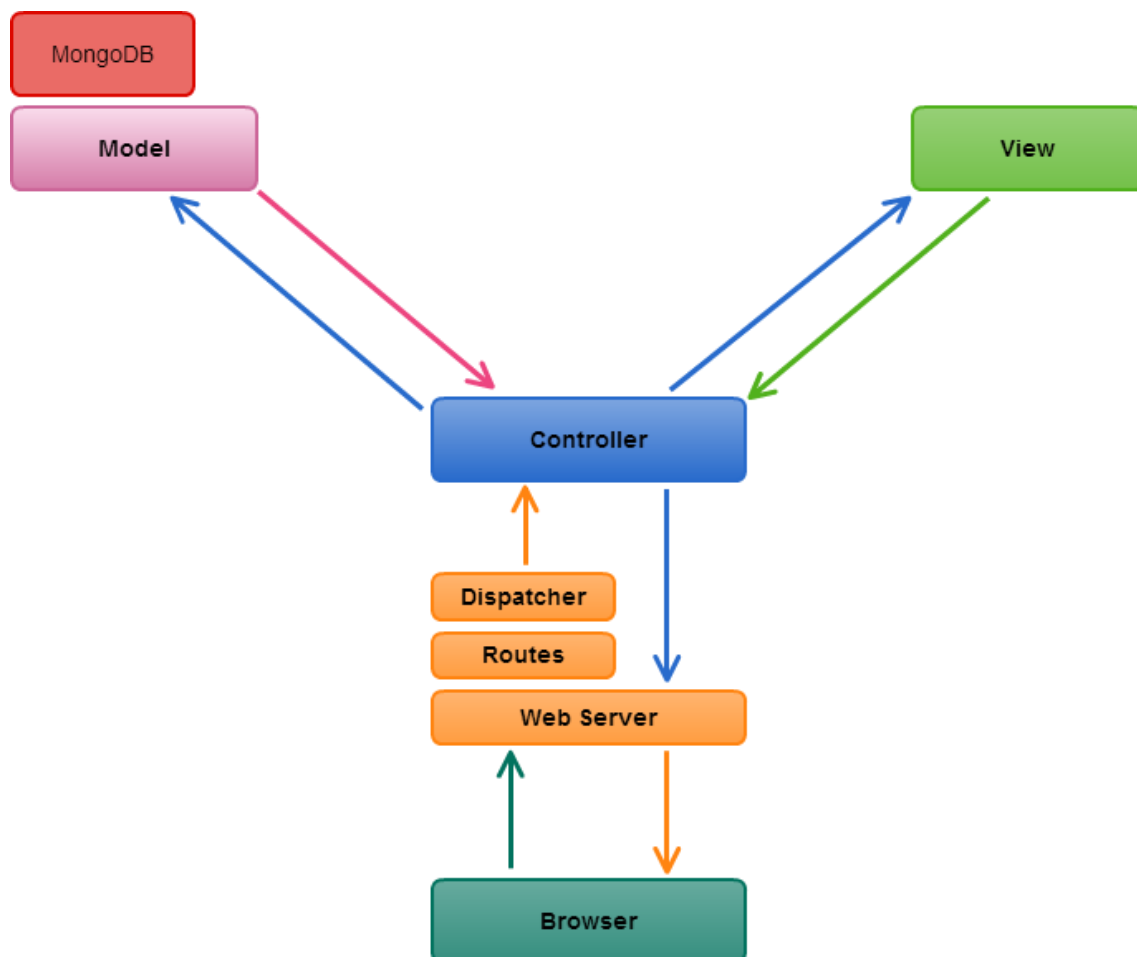


Figura 1: Diagrama de la arquitectura general de la plataforma



Arquitectura de la base de datos

La base de datos es un pilar fundamental de la aplicación puesto que es quien contiene toda la información relativa a cada uno de los componentes del sistema.

Como se ha explicado antes, la base de datos ha sido implementada utilizando MongoDB, una base de datos NoSQL basada en documentos JSON.

A continuación se expone el diagrama entidad-relación de la base de datos, donde se muestran los documentos JSON existentes y las relaciones entre los mismos. En el diagrama podemos distinguir los siguientes documentos JSON principales

- User: Guarda la información esencial de un usuario: su nombre, nick, password, email, etc.
- Resource: Guarda la información de cada uno de los recursos del sistema.
- Course: Contiene la información de cada uno de los cursos que han sido creados en el sistema.
- Lecture: En este documento se guardan la información relativa a los módulos del sistema.
- Lesson: Guarda la información relativa a las lecciones existentes en el sistema.
- Test: Contiene la información relativa a los test creados por los usuarios en el sistema.

Como se ha visto en los casos de uso, donde un curso contenía varios módulos que, a su vez, contenían varias lecciones, las cuales contenían varios recursos y éstos contenían varios test. Esta contención está representada en la base de datos mediante las relaciones existentes entre las tablas Course, Lecture, Lesson, Resource y Test. Siguiendo las líneas de relación se observa que:

- Un curso comprende varios módulos.
- Un módulo comprende varias lecciones.
- Una lección comprende varios recursos.
- Un recurso comprende varios test.

Esta cadena de relación es extrapolable al resto de relaciones existentes entre los casos de uso descritos anteriormente.

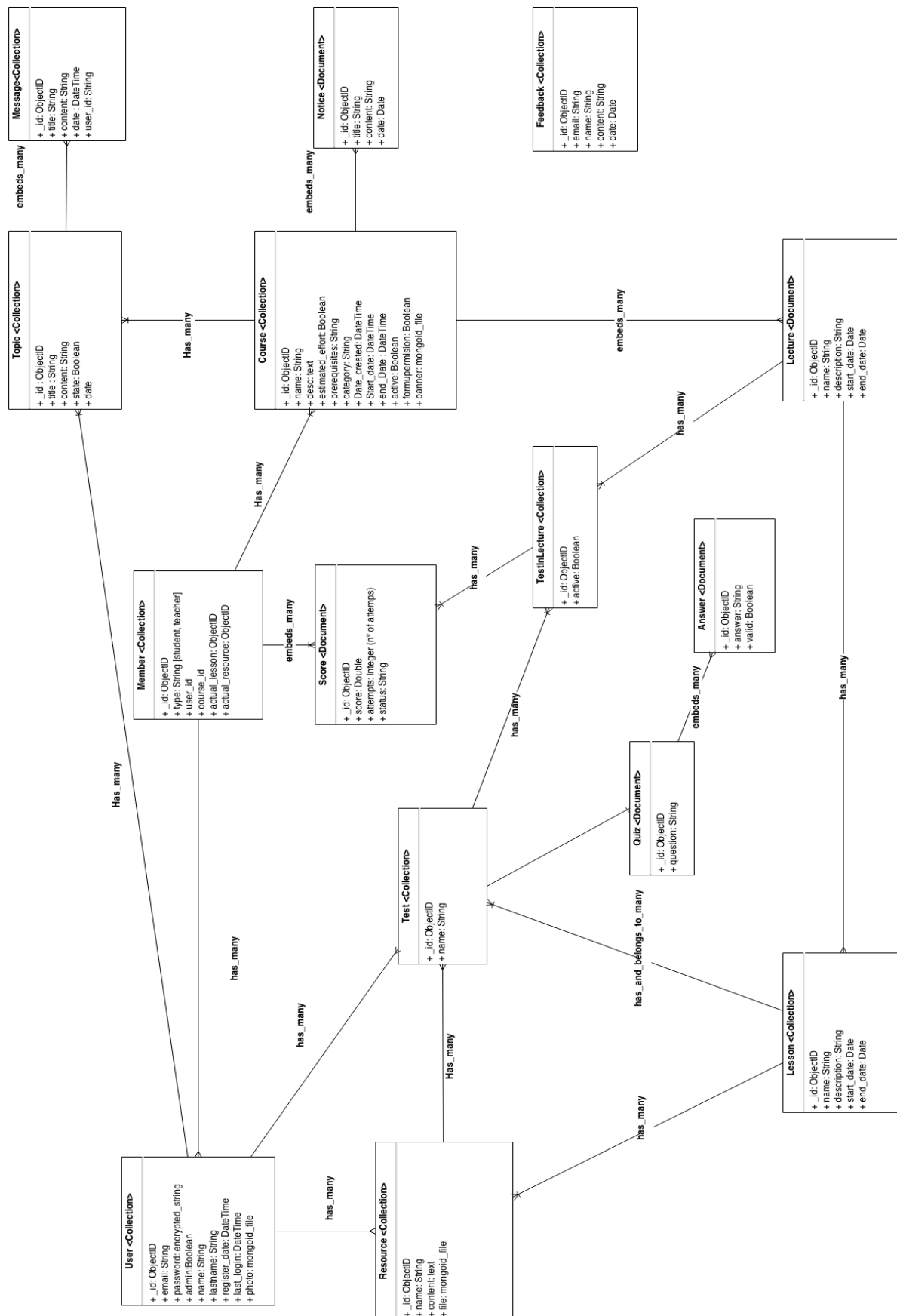


Figura 2: Diagrama Entidad-Relación de la base de datos.



Arquitectura de la aplicación

La aplicación se ha desarrollado siguiendo el patrón Modelo-Vista-Controlador (MVC), hacia el cual está orientado el propio desarrollo en Ruby on Rails.

MVC es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para conseguir esto, MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador.

Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, con lo que se busca facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento.

De manera genérica, los componentes de MVC se podrían definir de la siguiente manera:

- **El Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto las consultas como las actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación, es decir, en la lógica de negocio. El modelo envía a la vista aquella parte de la información que en cada momento le es solicitada para que sea mostrada. Las peticiones de acceso o manipulación llegan al modelo a través del controlador.
- **El Controlador:** Responde a eventos, normalmente invocados mediante acciones del usuario, e invoca peticiones al modelo cuando se hace alguna solicitud sobre la información, como puede ser la modificación de un registro sobre una base de datos. También envía comandos a su vista asociada si se solicita un cambio en la forma en la que se presenta el modelo, por ejemplo pasar de página en una tabla o documento mostrado. Por tanto se puede decir que el controlador hace de intermediario entre la vista y el modelo.

- **La Vista:** Presenta el modelo, es decir, la información y la lógica de negocio en un formato adecuado para interactuar, como puede ser una interfaz de usuario. Por tanto la vista requiere del modelo toda la información que debe representar como salida.

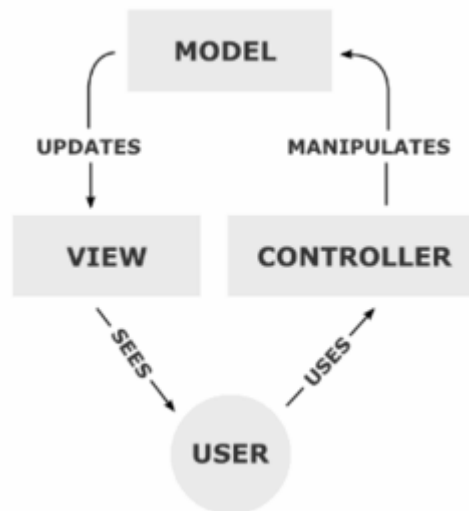


Figura 3: Diagrama general del patrón MVC.

El flujo de control básico que se lleva a cabo en una aplicación MVC es el siguiente:

1. El usuario interactúa con la interfaz de usuario de alguna manera, como puede ser pulsar un botón, escribir un texto, etc.
2. El controlador recibe, por parte de los objetos de la vista, la notificación de la acción solicitada por el usuario. El controlador gestiona el evento que llega mediante un gestor de eventos.
3. El controlador accede al modelo, posiblemente modificando el mismo de forma adecuada a la acción solicitada por el usuario. Los controladores complejos están a menudo estructurados usando un patrón de comando que encapsula las acciones y simplifica su extensión.



4. El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada al usuario donde se reflejan los cambios en el modelo. El modelo no debe tener conocimiento directo sobre la vista.
5. La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

Este ciclo se repite constantemente durante todo el uso de la aplicación. Además, es adaptable para un correcto uso en cada desarrollo, como se ha visto en el capítulo “Arquitectura general del sistema”, donde se ha adaptado para su uso con Ruby on Rails y MongoDB.



Capítulo 6:

Trabajo futuro



La aplicación se ha pretendido entregar lo más completa posible en función del tiempo disponible para su desarrollo. Aun así se proponen distintas alternativas orientadas a un trabajo futuro sobre la plataforma para completarla y hacer de la misma una aplicación lo más potente posible y con mayor cantidad de componentes orientados al aprendizaje gratuito, compartido y comunitario.

- **Chat:** Se propone realizar una extensión de la plataforma donde se permitan salas de chat entre usuarios que sean amigos, usuarios suscritos al mismo curso que se encuentren conectados al mismo tiempo, profesores que compartan categoría de cursos, etc.
- **Red Social:** Se propone una extensión que funcione como red social, donde los usuarios compartan peticiones de amistad entre ellos y mantengan una lista de amigos con los que podrán intercambiar mensajes e invitaciones a otros cursos, o participar conjuntamente en la creación de nuevos cursos.
- **Diplomas:** Se propone una extensión que funcione como generador de diplomas no oficiales al término de un curso.
- **Sistema de logros:** Se propone una extensión mediante la cual se implemente un sistema de logros, de tal manera que los usuarios acumulen puntos y logros según la cantidad de cursos que completen, la calificación que consigan en los mismos, la participación que ofrezcan en la creación de cursos, etcétera.

Por otro lado, se intentará dar a la plataforma una publicidad orientada al uso de la misma por organismos universitarios, puesto que el proyecto se ha orientado hacia ese tipo de usuarios.

De esta manera, se ofrecerá a diversas universidades la plataforma para su implantación y uso, así como jornadas de formación orientadas a ampliar la generalización y usabilidad de la misma.



Capítulo 7:

Conclusiones



Una vez finalizado el desarrollo del proyecto se ha obtenido una plataforma de MOOC de código abierto, robusta, ligera, estable y con una novedosa interfaz de usuario fácil e intuitiva para conseguir una mejor experiencia por parte de los usuarios, tanto los habituales como los nuevos.

Las principales novedades a destacar en la aplicación desarrollada que la distinguen de las actuales son, a nuestro parecer, tres:

- **Desarrollo en Ruby on Rails:** La aplicación ha sido desarrollada con un lenguaje sencillo pero a la vez potente, open source y novedoso.
- **Utilización de MongoDB:** La aplicación no utiliza las bases de datos SQL habituales, sino que hemos apostado por una base de datos NoSQL y open source que actualmente se encuentra en constante auge.
- **Un usuario como profesor y alumno:** La plataforma ofrece la posibilidad a los usuarios de compartir su conocimiento a la vez que aprenden nuevas materias.

El desarrollo del proyecto ha sido complicado por dos motivos principales: la utilización de nuevas tecnologías innovadoras y desconocidas para nosotros, así como la incompatibilidad de horarios por motivos laborales y personales.

Gracias a una buena coordinación entre los miembros del equipo hemos podido superar las incompatibilidades de horarios y repartirnos las tareas según la disponibilidad de cada uno, aprendiendo en el camino el uso de herramientas de gestión de proyectos y tareas como Producteev y Trello.

Por otra parte, gracias a la amplia documentación existente en internet y en las bibliotecas, hemos aprendido cómo diseñar, manejar e implementar las nuevas tecnologías que componen la plataforma descrita en el presente documento.

Como conclusión final, este proyecto de Sistemas Informáticos ha supuesto un reto y un aprendizaje muy valioso para nuestro desarrollo personal y profesional, en el cual hemos aprendido muchas cosas que, muy probablemente, utilizaremos en un futuro cercano.

Por supuesto agradecer a D. Antonio Sarasa su tutela durante todo el curso académico y el desarrollo del proyecto, así como la libertad en la decisión del tema del mismo y de las tecnologías a utilizar. Sus consejos y seguimiento continuos han servido para finalizar la plataforma y, sobre todo, para obtener nuevos y valiosos conocimientos.



Bibliografía

- Agile Web Development with Rails 4.
Ruby, Sam; Thomas, Dave; Heinemeier Hansson, David.
Edited by Susannah Davidson Pfalzer.
- Everyday Rails Testing with RSpec.
Sumner, Aaron.
- Ruby on Rails Guides (v4.1.1).
<http://guides.rubyonrails.org/>
- Getting Started with Rails 4.x on Heroku.
<https://devcenter.heroku.com/articles/getting-started-with-rails4>
- The MongoDB 2.6 Manual.
<http://docs.mongodb.org/manual/>
- Bootswatch Documentation.
<http://bootswatch.com/flatly/>
- Highcharts Documentation.
<http://www.highcharts.com/docs>
- Mongoid Documentation.
<http://mongoid.org/en/mongoid/index.html>
- Mongolab Documentation.
<http://docs.mongolab.com/>
- <http://es.wikipedia.org>



Autorización a la UCM

Los siguientes alumnos matriculados en la Facultad de Informática de la Universidad Complutense de Madrid autorizan a las mismas a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a los autores, tanto la propia memoria, como el código, los contenidos audiovisuales incluso si incluyen imágenes de los autores, la documentación y/o el prototipo desarrollado.

Alejandro Armenteros Rodelgo
53474222G

Cristian García García
70823957B

José Félix Martín Ramírez
45852254Y



Anexos

- Anexo I: Manual de Usuario de la plataforma.
- Anexo II: Manual de Instalación de la plataforma UCMoocs en un servidor Debian/Ubuntu.
- Anexo III: Manual de despliegue de la plataforma UCMoocs en Heroku.



Universidad Complutense

UCMoocs

Manual de Usuario

Martín, Felix

García, Cristian

Armenteros, Alejandro



2014

Índice

- **Índice**
- **Introducción.**
- **Conceptos básicos.**
- **Paseo por la aplicación.**
 - **Primeras impresiones.**
 - Página principal
 - Registro
 - Login
 - Perfil
 - Feedback
 - Barra superior
 - Cursos
 - **Crear un curso como profesor.**
 - Crear un curso.
 - Gestionar recursos.
 - Gestionar tests.
 - Lecturas.
 - Lecciones.
 - **Inscribirse en un curso como alumno.**
 - Ver las lecturas.
 - Ver las lecciones.
 - Ver los tests.
 - **Durante el curso.**
 - Noticias
 - Foro.
 - Evaluación.
 - Seguimiento.
- **FAQ.**

Introducción

Un **MOOC** es una modalidad de educación abierta que consiste en cursos ofrecidos gratuitamente a través de plataformas educativas en Internet cuya filosofía es la liberación del conocimiento para que este llegue a un público más amplio.

Para que la enseñanza a distancia pueda ser considerada MOOC debe cumplir los siguientes requisitos:

- **Ser un curso:** Debe contar con una estructura orientada al aprendizaje, que suele conllevar una serie de pruebas o evaluaciones para acreditar el conocimiento adquirido.
- **Tener carácter masivo:** El número de posibles matriculados es, en principio, ilimitado; o bien en una cantidad muy superior a la que podría contarse en un curso presencial. El alcance es global.
- **En línea:** El curso es a distancia e Internet es el principal medio de comunicación. No requiere la asistencia a un aula.
- **Abierto:** Los materiales son accesibles de forma gratuita en Internet. Ello no implica que puedan ser reutilizados en otros cursos.

Este manual está dirigido a que los usuarios aprendan a utilizar la plataforma **UCMoocs**. Puede acceder a esta plataforma mediante la siguiente URL.

<http://moocs.herokuapp.com/>

Conceptos básicos

Esta sección explicara conceptos que serán útiles para poder entender este manual.

Feedback: Es una palabra inglesa que significa **retroalimentación** o **dar respuesta** a un determinado pedido o acontecimiento.

Reportar: Significa comunicar, transmitir o notificar algo a alguien.

Foro: Es un sistema que da soporte a opinión o discusiones en línea.

Topic: Es cada uno de los temas en los que se estructura el foro.

Paseo por la aplicación

Vamos a ir punto por punto viendo las opciones que tiene esta plataforma. El fin de esta sección es que el usuario se familiarice con todas las secciones de UCMoocs.

Primeras impresiones.

La primera visión que tendremos de la aplicación es la de la página principal.



Figura 1

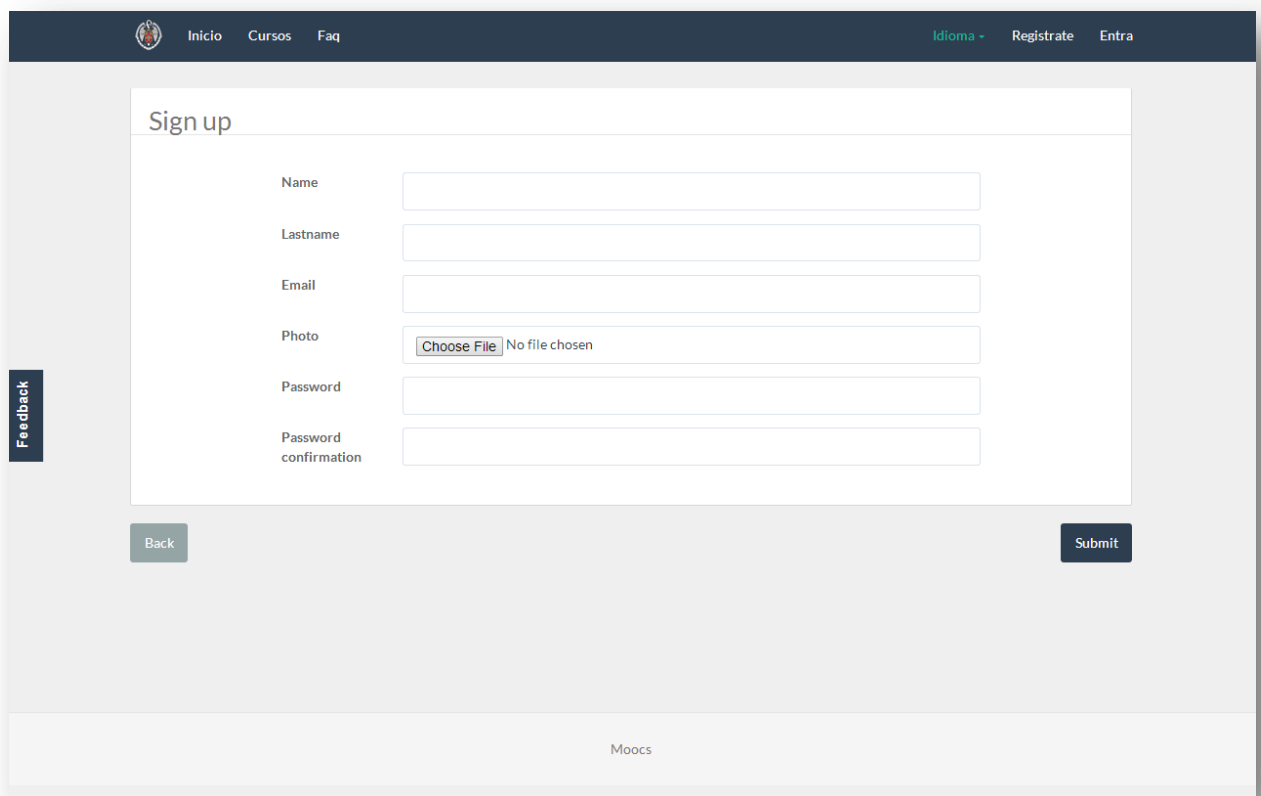
En esta ventana principal podemos ver:

- **La barra principal.** La cual estará siempre accesible, y nos permitirá una navegación rápida entre distintas partes de la aplicación, además de la opción de cambiar el idioma.
- **El botón de feedback.** Este botón nos permite abrir la ventana de reportes, para reportar cualquier fallo o mejora para la plataforma. Estará visible en toda la plataforma.

- **El buscador.** Facilita la búsqueda de cursos concretos, haciendo una búsqueda parcial o total del nombre de un curso.
- **Cursos cercanos.** Aquí veremos un listado de los cursos que tienen un comienzo cercano. Lo cual nos permite estar al día de los cursos más próximos.

Registro

El primer paso para utilizar la aplicación es registrarse. Para esto tenemos en cuenta que aún no tienes una cuenta creada en la plataforma. Accedemos al registro con la pestaña registrarse de la barra superior.



The screenshot shows the 'Sign up' registration form. The form is titled 'Sign up' and contains the following fields: Name, Lastname, Email, Photo, Password, and Password confirmation. The Photo field has a 'Choose File' button and the text 'No file chosen'. Below the form are 'Back' and 'Submit' buttons. The top navigation bar includes 'Inicio', 'Cursos', 'Faq', 'Idioma', 'Registrate', and 'Entra'. A 'Feedback' button is visible on the left side of the form area. The footer of the page shows 'Moocs'.

Figura 2

Debemos rellenar los campos, siendo todos obligatorios a excepción de la foto.

Una vez enviada y comprobada la información, su perfil de usuario pasara a formar parte de la base de datos.

Login

Si por otro lado, ya tiene creada una cuenta en la plataforma, puede acceder al login presionando en la pestaña Entrar del menú superior.

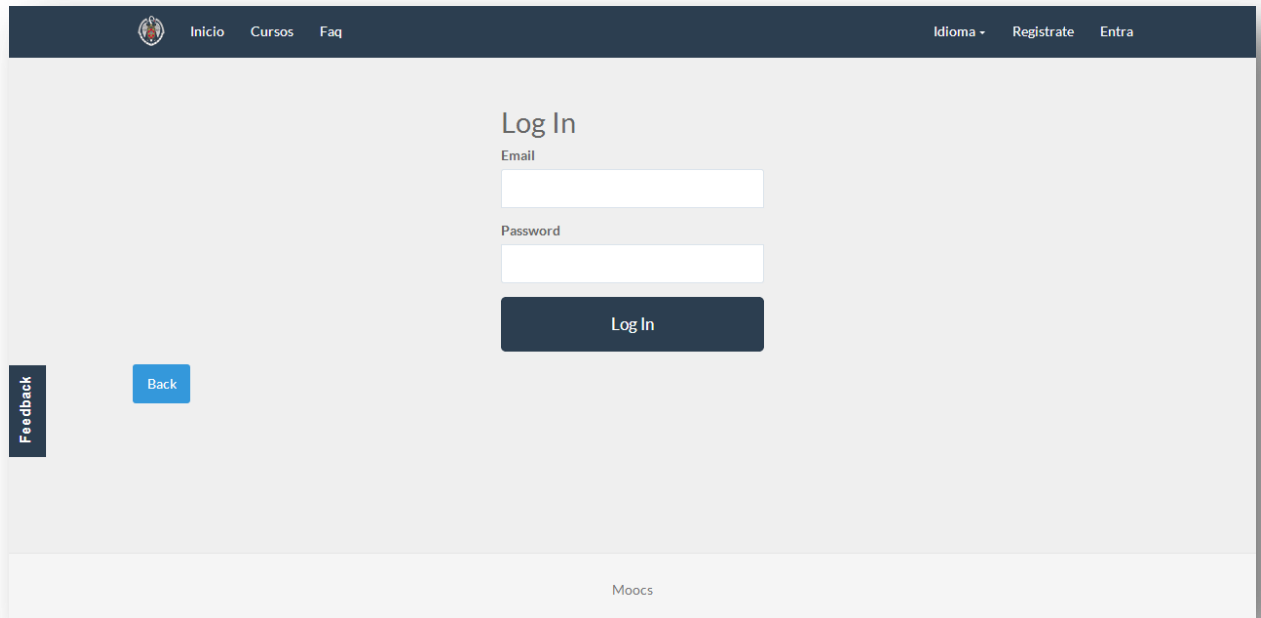
The image shows a web browser window displaying the login page of the UCMoocs platform. The page has a dark blue header with the university's logo and navigation links: 'Inicio', 'Cursos', 'Faq', 'Idioma', 'Registrate', and 'Entra'. The main content area is light gray and features a 'Log In' section with two input fields for 'Email' and 'Password', and a dark blue 'Log In' button. On the left side, there is a vertical 'Feedback' button and a 'Back' button. The footer is white and contains the word 'Moocs'.

Figura 3

Simplemente introduzca sus datos en los campos de texto y acceda a la aplicación.

Sea cual sea el camino, ya hemos accedido a la aplicación y tenemos nuestro usuario disponible en la plataforma.

Barra superior

Uno de los elementos más importantes de la navegación es la Barra superior. Al estar logueados con nuestro usuario veremos la barra de la siguiente forma.

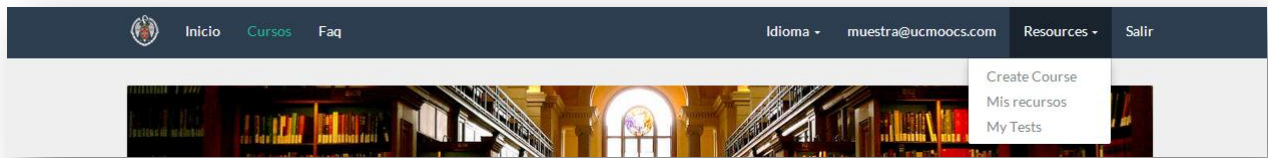


Figura 4

Esta barra se compone, de izquierda a derecha, de las siguientes opciones:

- **Complutense.** Puede acceder a la web de la complutense pulsando en el icono.
- **Inicio.** Esta opción nos llevara a la página principal.
- **Cursos.** Esta pestaña nos llevara al menú de los cursos.
- **FAQ.** En esta sección podemos ver las preguntas frecuentes de los usuarios.
- **Idioma.** Es un desplegable que nos permitirá cambiar el idioma de la aplicación. Actualmente están habilitados el español e inglés.
- **Perfil.** A continuación se nos muestra el correo asociado a nuestra cuenta asociada a la plataforma. Pulsar en este correo no lleva a la página de perfil.
- **Resources.** Mediante este desplegable podemos:
 - Crear un curso nuevo.
 - Gestionar los recursos.
 - Gestionar los tests.
- **Salir.** Cerraremos nuestra sesión de UCMoocs.

Perfil

En el perfil podemos ver los cursos en los que estamos matriculados, para así poder acceder fácilmente a los diferentes cursos.

Además se nos muestra la foto y correo asociados a nuestro perfil.

Podemos editar tanto estos parámetros, como otros referentes al perfil de usuario, como la contraseña, en el botón editar perfil.

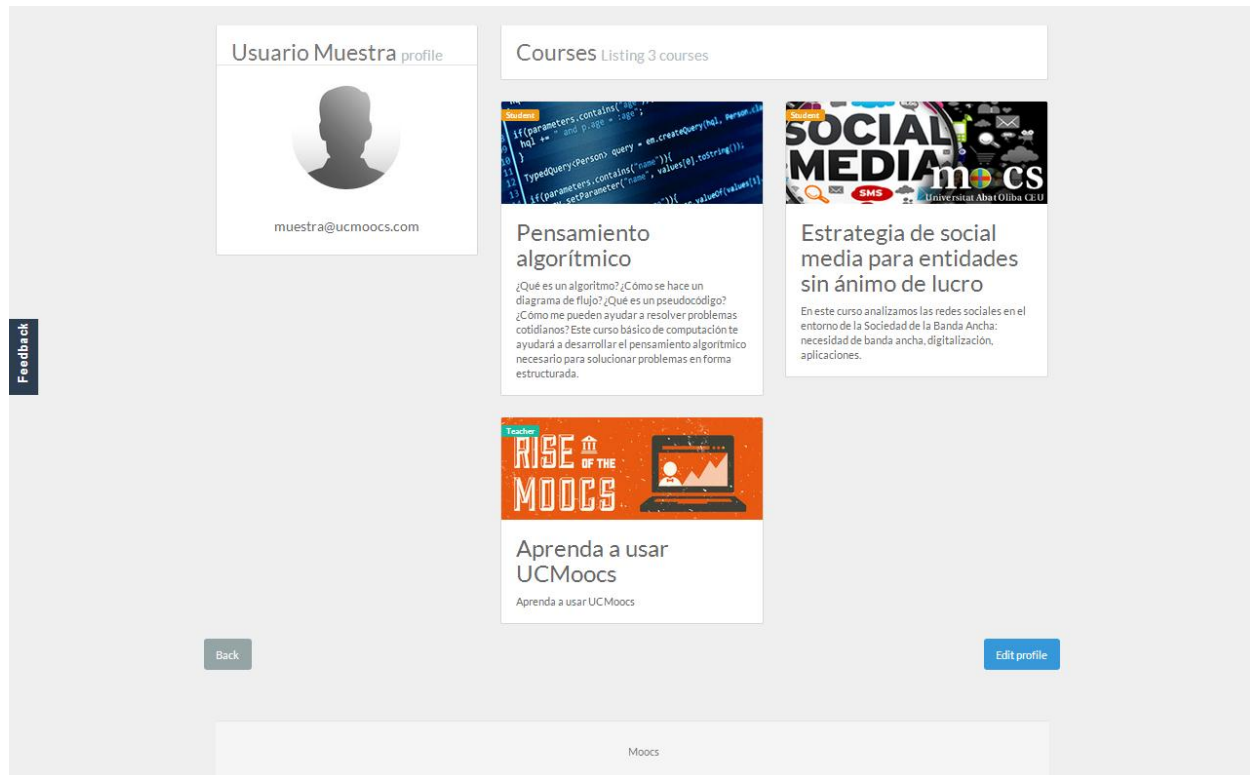


Figura 5

Este perfil será nuestra identificación en la aplicación y todo lo que hagamos será asignado a este perfil.

Feedback

Esta será una de las formas en las que el usuario podrá comunicarse con los administradores y desarrolladores, con el fin de ayudar al mantenimiento, mejora y desarrollo de la plataforma.

La opinión del usuario es parte importante de la mejora de la aplicación. Toda sugerencia e idea que nos ayude a mejorar es bienvenida.

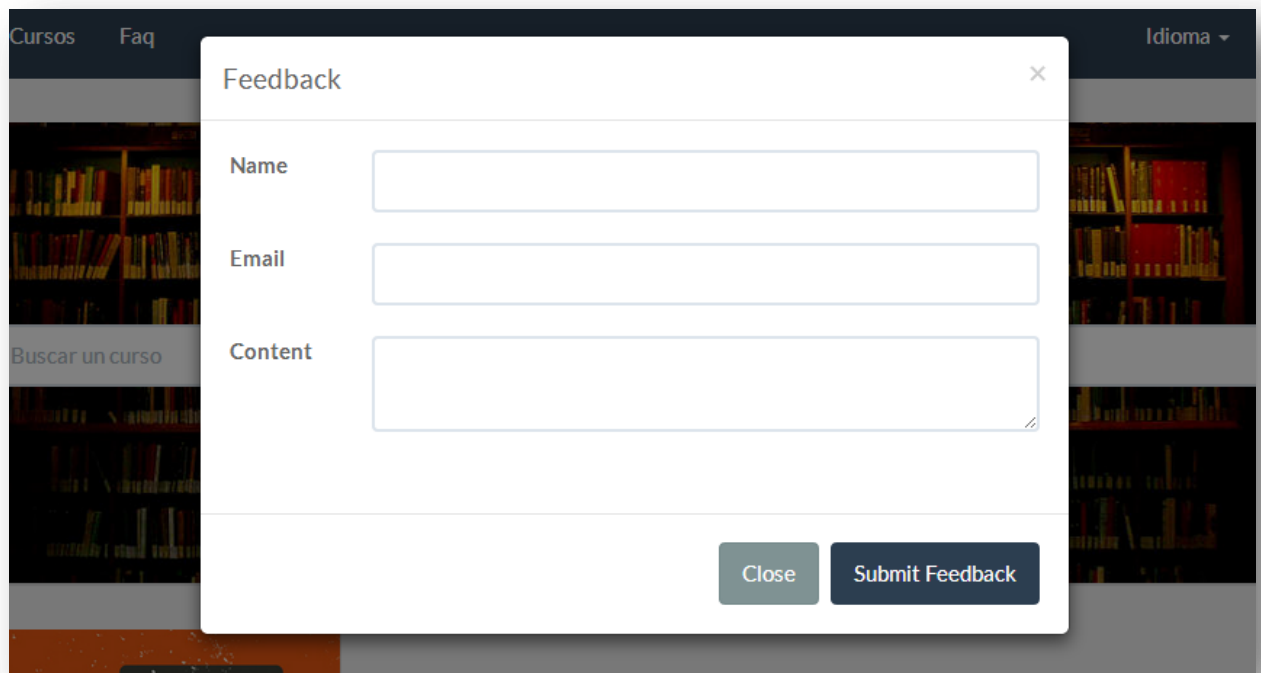
The image shows a screenshot of the UCMoocs website interface. A 'Feedback' modal window is open in the center. The modal has a title bar with 'Feedback' and a close button (X). It contains three input fields: 'Name', 'Email', and 'Content'. The 'Content' field is a larger text area. At the bottom right of the modal are two buttons: 'Close' and 'Submit Feedback'. The background of the website is visible, showing a dark header with 'Cursos', 'Faq', and 'Idioma' links, and a sidebar with a search bar labeled 'Buscar un curso'.

Figura 6

Cursos

En el menú de cursos, al que podemos acceder desde la barra superior, se muestra las categorías de los cursos actuales, con el número de cursos en cada categoría.

Además, nada más entrar en este apartado, en el menú derecho podremos ver los cursos más populares. Y una vez seleccionada categoría, los cursos asociados a dicha categoría.

Este panel puede verse independientemente de si se esta logueado o no en la aplicación.

Las categorías que hemos seleccionado son las siguientes:

- Arte
- Biología y ciencias naturales
- Administración de empresas
- Química
- Ciencias de la computación: Inteligencia artificial
- Ciencias de la computación: Ingeniería del software
- Ciencias de la computación: Sistemas y Seguridad
- Ciencias de la computación: Teoría
- Educación

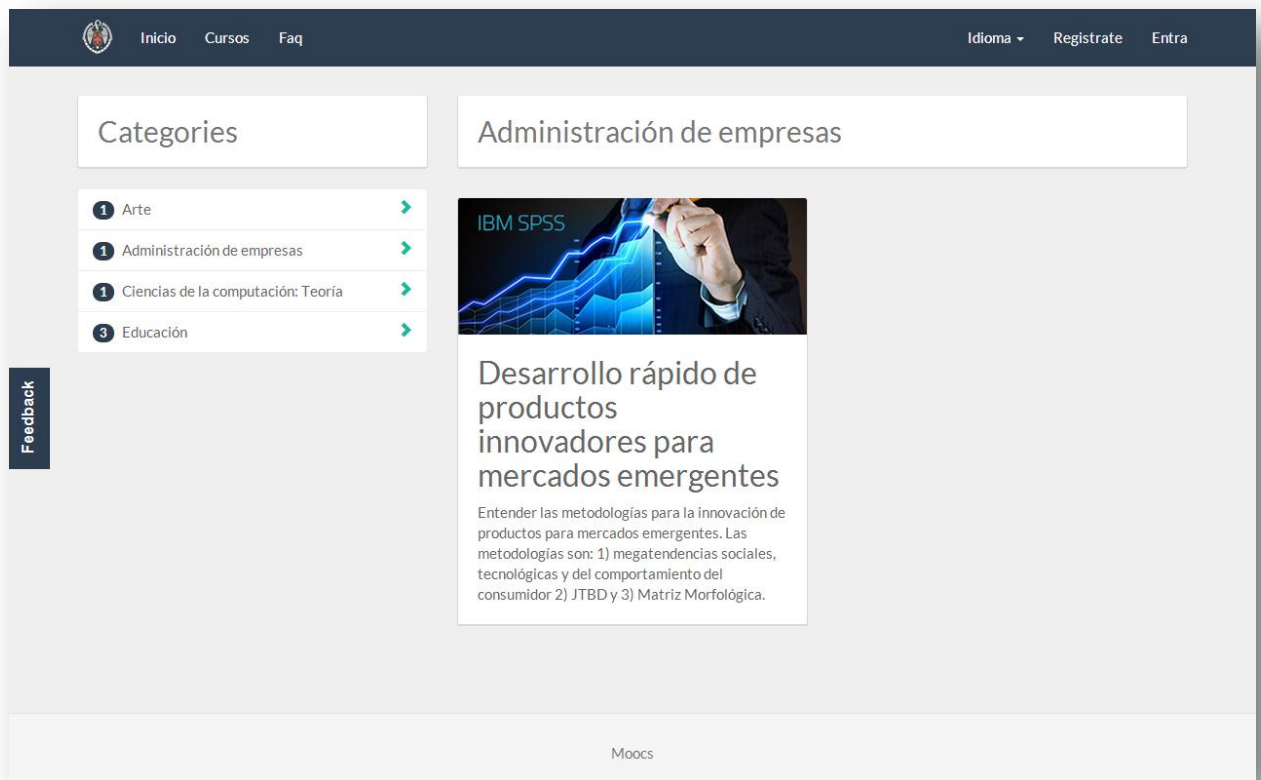


Figura 7

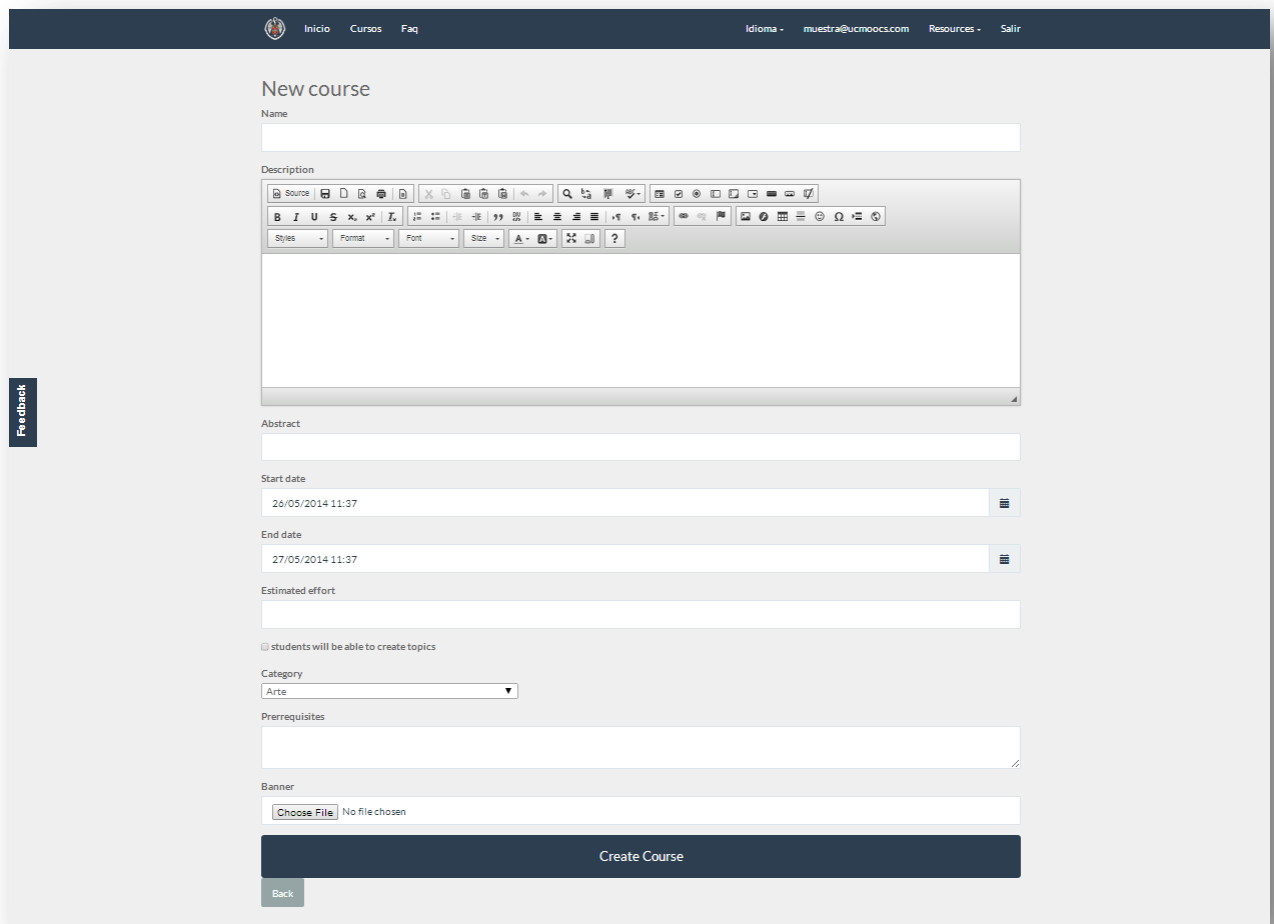
Ser profesor de tus propios cursos

Este apartado del manual va dirigido a aquellos usuarios que quieren crear sus cursos, y hacer las veces de profesor para todos los usuarios que estén interesados en inscribirse en el curso.

Por lo tanto lo primero y más importante a la hora de tener tu curso es: Crear tu curso.

Crear un curso

Usamos la opción del desplegable Recursos. Al pulsar sobre crear curso nos aparecerá un formulario donde podremos completar los datos más importantes de nuestro curso.



The screenshot shows the 'New course' form in the UCMoocs system. The form is titled 'New course' and contains the following fields and options:

- Name:** A text input field.
- Description:** A rich text editor with a toolbar containing various icons for text formatting, alignment, and insertion.
- Abstract:** A text input field.
- Start date:** A date and time picker showing '26/05/2014 11:37'.
- End date:** A date and time picker showing '27/05/2014 11:37'.
- Estimated effort:** A text input field.
- Students will be able to create topics:** A checkbox that is currently unchecked.
- Category:** A dropdown menu with 'Arte' selected.
- Prerequisites:** A text input field.
- Banner:** A file upload field with a 'Choose File' button and the text 'No file chosen'.
- Create Course:** A large blue button at the bottom of the form.
- Back:** A small button at the bottom left of the form.

On the left side of the form, there is a vertical 'Feedback' button.

Figura 8

En este formulario deberá rellenarse la información necesaria sobre todo lo que va a ser nuestro curso. No hay que preocuparse, ya que una vez creado podremos editar este contenido.

El crear el curso nos llevara a la vista principal del profesor del curso. En esta vista podemos ver toda la información que hemos rellenado en el formulario.

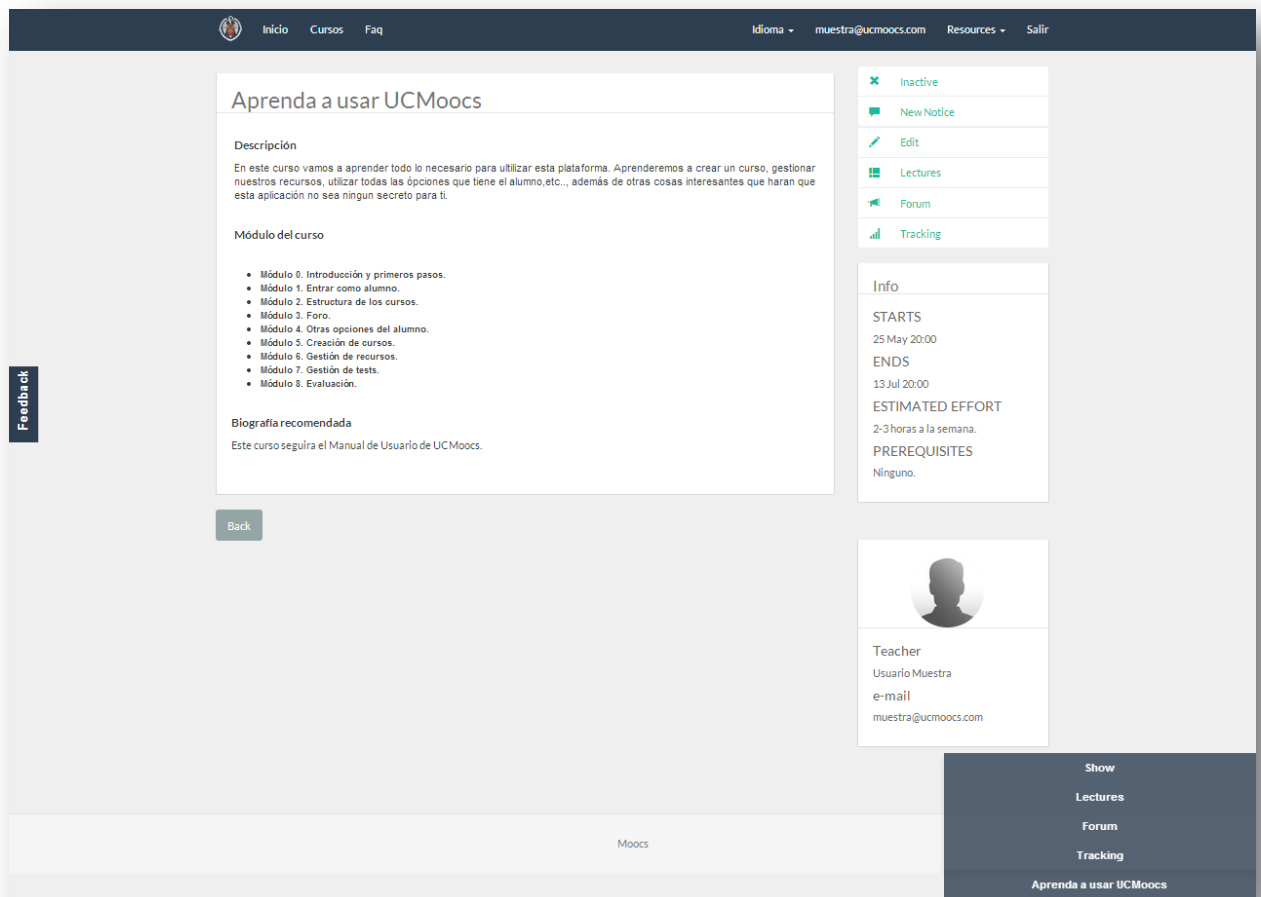


Figura 9

Además, esta vista nos ofrece diferentes opciones:

- **Activar/desactivar.** Esta pestaña hace que podamos activar o desactivar el curso. El curso no será visible por los usuarios hasta que se active.
- **Nueva noticia.** Estas noticias serán lo primero que vean los usuarios al entrar al curso.
- **Editar.** Podemos editar los datos del curso.
- **Lecturas.** Accedemos a las lecturas o módulos del curso.
- **Foro.** Para acceder al foro del curso.

- **Seguimiento.** En esta pestaña podremos ver la evolución de los alumnos en nuestro curso.
- **Navegación rápida.** Esta opción estará abierta durante la navegación del curso, y te permitirá ir de forma rápida de una sección a otra.

Gestionar recursos

Podemos acceder a la gestión de recursos mediante la opción mis recursos del menú recursos situado en la barra superior.

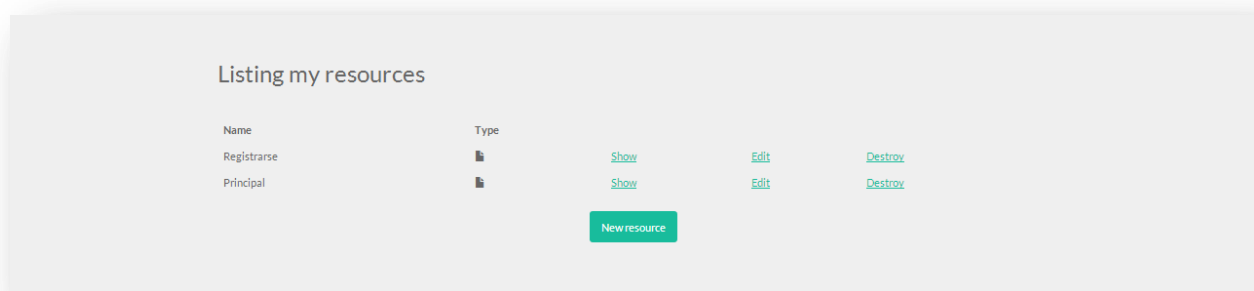


Figura 10

Estos cursos nos permitirán añadir contenido a nuestro curso.

Los recursos son propios del usuario, y permanecerán guardados aun si no están en ningún curso.

Figura 11

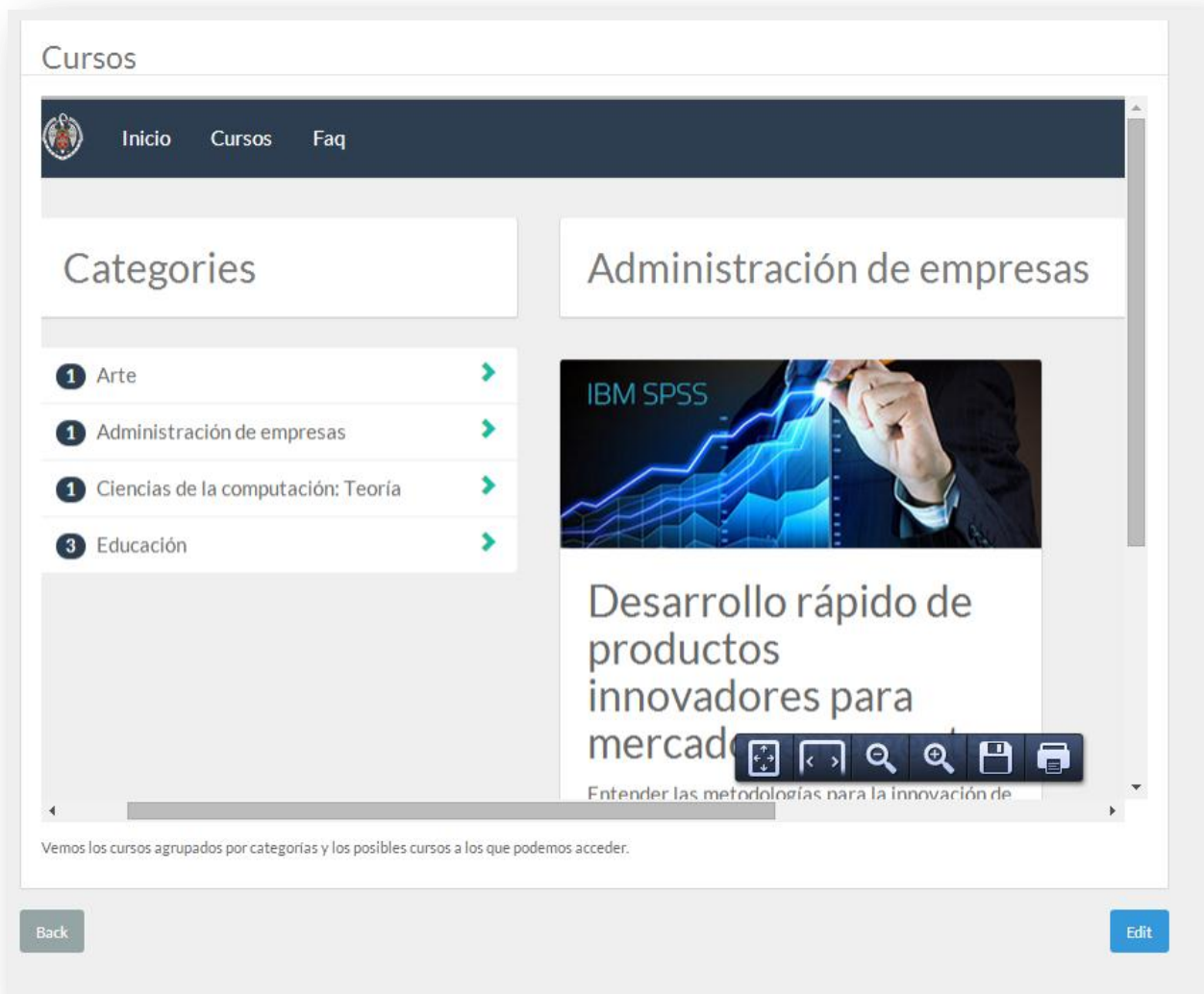


Figura 12

Podemos gestionar nuestros recursos al gusto, sabiendo que tenemos unas limitaciones en el tipo de datos que usamos.

Actualmente las extensiones permitidas son: mp4, mp3 y pdf.

Gestionar tests

A la gestión de tests se accede de forma similar a la gestión de recursos.

Pero la creación de los mismos es muy diferente.

New quiz

Name
Cursos

Question

¿Un curso puede crearse por un usuario no registrado?

body p

Answer Valid ☐

Back add answer

Answer	Valid	Delete
No	✗	+
Sí	✓	+

Submit

Moocs

A la izquierda vemos el nombre que le asignamos, y la pregunta. Además, para añadir respuestas basta con meterlas en la caja de texto respuesta, marcar si es correcta o no, y pulsar en añadir pregunta.

A la derecha podemos ver las preguntas que hemos añadido, y borrarlas mediante el botón de borrar de cada una de ellas.

Podemos crear indistintamente tests de respuesta única, y tests multirespuesta.

Lecturas

Los cursos están separados en lecturas o módulos. Los cuales se usan para separar y estructurar el contenido del curso.

En esta vista se lista los módulos del curso, indicando cual es el que está actualmente activo. Y al ser el profesor, dando las posibilidades de edición del módulo.

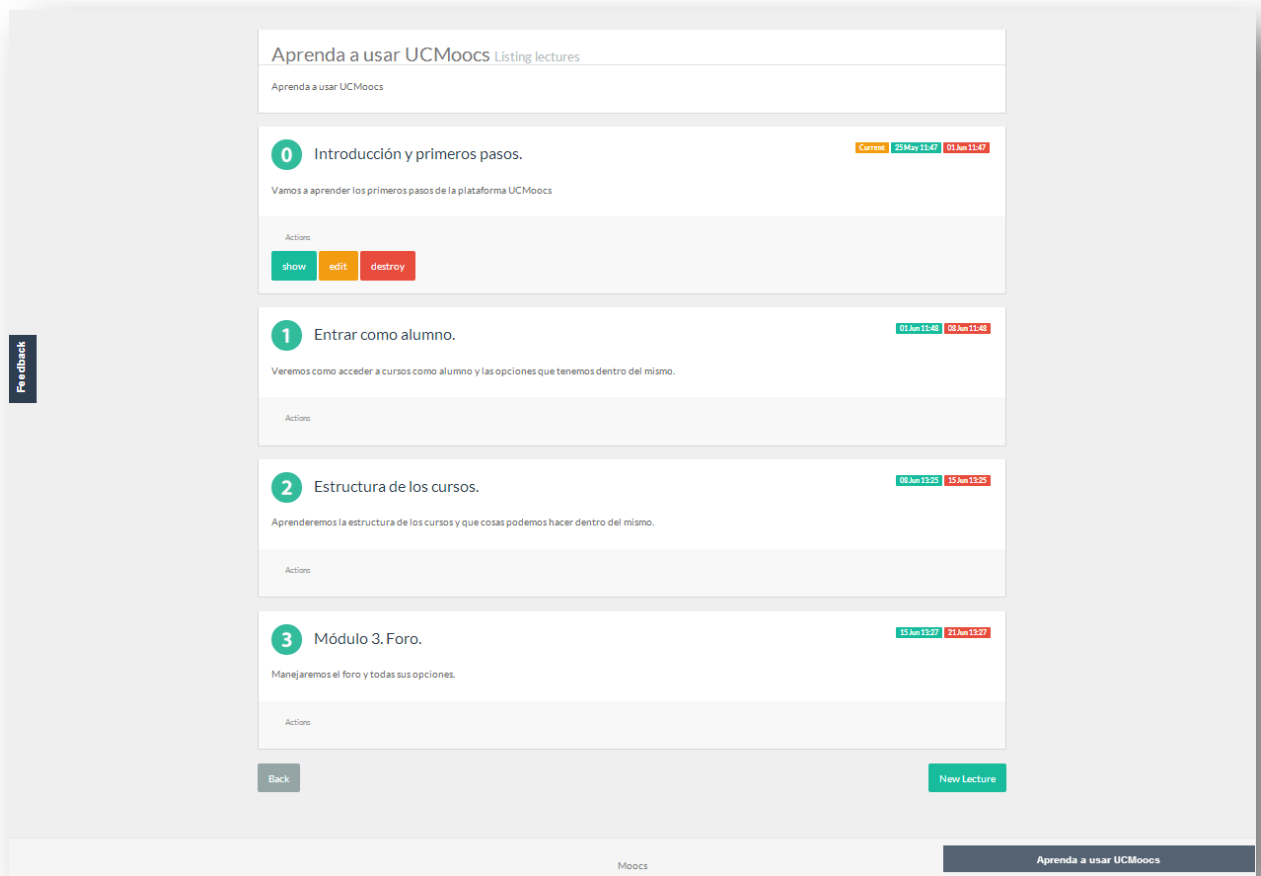


Figura 13

Aquí podemos añadir más lecturas, o entrar en alguna de las ya creadas.

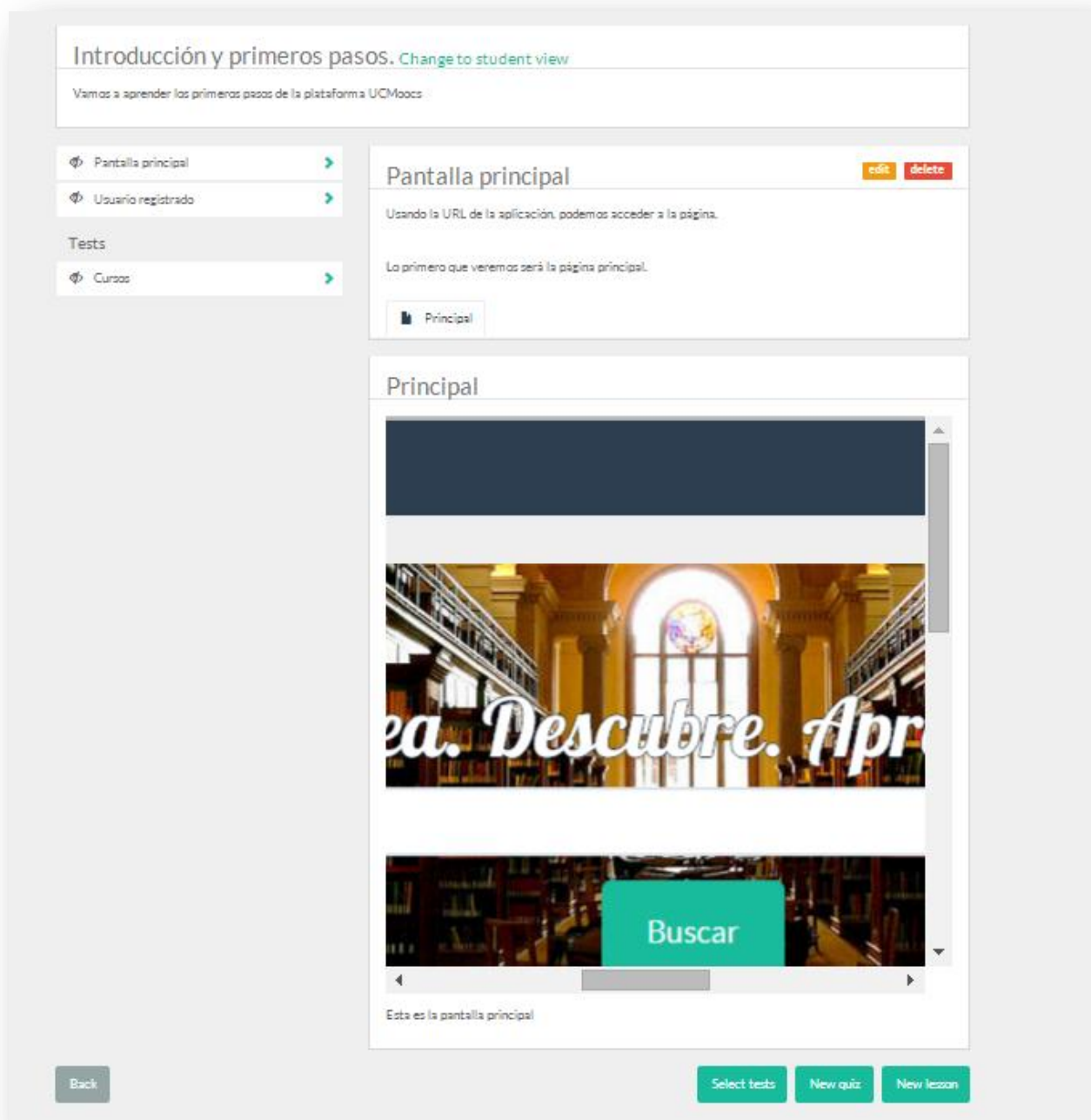


Figura 14

Esta pantalla ofrece muchas posibilidades.

Por un lado tenemos las lecciones y tests asociados a un curso. Y en el lado derecho, tenemos la lección o test seleccionado.

Por otro lado vemos la posibilidad de añadir tests o lecciones o añadir test ya creados.

Los tests creados de esta manera no se borrarán si se borra el curso, si no que serán reutilizables por el profesor.

Además, los tests que añadimos o creamos de esta forma, se asignan al módulo y son evaluables.

Aparte de estos test, los módulos están compuestos por lecciones.

Lecciones

Las lecciones son la última parte que nos queda por ver de la estructura del curso.

A estas lecciones las adjudicaremos recursos y tests.

Los recursos pueden ser tanto creados como nuevos, y ambos seguirán existiendo si la lección desapareciera.

En la izquierda tenemos los parámetros básicos de la lección. Mientras que a la derecha podemos gestionar los recursos y tests que incluiremos en la lección.

Hay que remarcar que estos tests asociados a lecciones no son evaluables, y su fin es, únicamente, afianzar los conceptos aprendidos sobre la lección.

Aprenda a usar UCMoocs Introducción y primeros pasos.

Edit Lesson

Basic Data

Name

Usuario registrado

Description

Un usuario registrado puede ser tanto profesor como alumno de un curso.

Para registrarse basta con rellenar el fácil formulario que vamos en la pestaña [Registrarse](#).

Un usuario no registrado puede ver los cursos, pero nunca podrá crearlos, ni inscribirse en ninguno, ni acceder a su contenido.

Back

Submit

Selected Resources

Registrarse

Usuario registrado

Select resources New Resource

Figura 15

Como profesor, no habrá ningún problema a la hora de modificar las lecciones en cualquier momento del curso.

Los tests los seleccionaremos con el botón de selección de recursos.

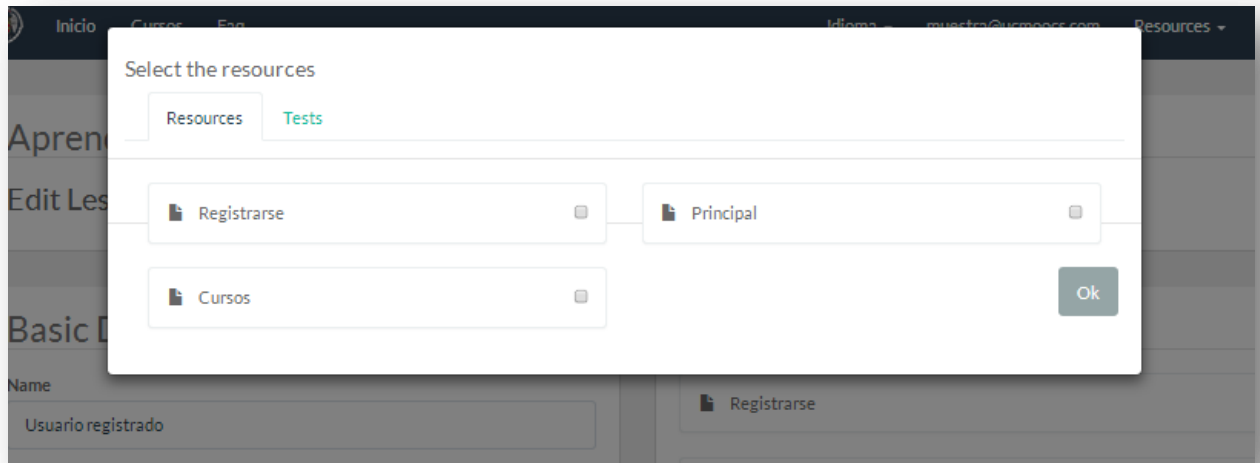


Figura 16

Con esta gestión de lecciones, ahora ya tenemos todos los componentes necesarios para crear nuestro curso.

Inscribirse en un curso como alumno.

En este apartado, vamos a tratar todo lo referente a la inscripción de cursos, y a la vista general del curso. Se trata solamente de lo referente a la visión de la estructura. Más adelante trataremos las opciones de los cursos durante los mismos.

Las opciones en el curso se limitan, pudiendo acceder a los módulos, al foro y al seguimiento (estos dos últimos se comentaran más adelante).

The screenshot displays the course interface for "Estrategia de social media para entidades sin ánimo de lucro". The main content area is divided into two sections: "Descripción" and "Módulos del curso".

Descripción: This section provides an overview of the course, explaining that it analyzes social networks in the context of the "Sociedad de la Banda Ancha" (Broadband Society), discussing the need for broadband, digitalization, and applications. It mentions that the course covers various denominations of the phenomenon, with a focus on digitalization and the need for broadband. The course aims to provide information and knowledge, with a key focus on digitalization and the need for broadband. It also mentions that the course will cover the construction of a website for the entity and a linked blog, explaining techniques (SEO, SEM, SMM) that can be used to obtain benefit for the entity. The goal is to achieve this without payment (SEO Y SIM). The course will explain strategies and provide exercises on the subject.

Módulos del curso: This section lists the course modules, each with a numbered icon and a title:

- Módulo 0. Presentación
- Módulo 1. Los social media en la era de la Sociedad de la Banda Ancha y el cloud journalism
- Módulo 2. Estrategias de comunicación en social media
- Módulo 3. Estrategias de negocio en social media
- Módulo 4. Análisis de casos
- Módulo 5. Repaso de conceptos básicos

Right Sidebar: This section contains additional course information:

- Lectures:** A list of lecture topics.
- Forum:** A link to the course forum.
- Tracking:** A link to track course progress.
- Info:** A section containing course details:
 - STARTS:** 18 May 08:00
 - ENDS:** 29 Jun 08:00
 - ESTIMATED EFFORT:** 4-6 horas de trabajo / semana
 - PREREQUISITES:** Banda Ancha
- Teacher:** A section containing the teacher's name and contact information:
 - Teacher:** test moocs
 - e-mail:** test.moocs@example.com

Bottom Bar: A dark blue bar at the bottom of the page contains a "Back" button on the left and the course title "Estrategia de social media para entidades sin ánimo de lucro" on the right.

Figura 17

Es importante comentar que el panel de navegación rápida estará disponible durante la navegación en todo el curso.

Ver las lecturas

En la visión de las lecturas tenemos ciertas limitaciones:

- Solo veremos las activadas por el profesor.
- Solo podremos entrar en las que ya se hayan impartido o en el módulo actual. Es decir, solo podremos ver aquellos de fechas actuales o pasadas, nunca los siguientes.

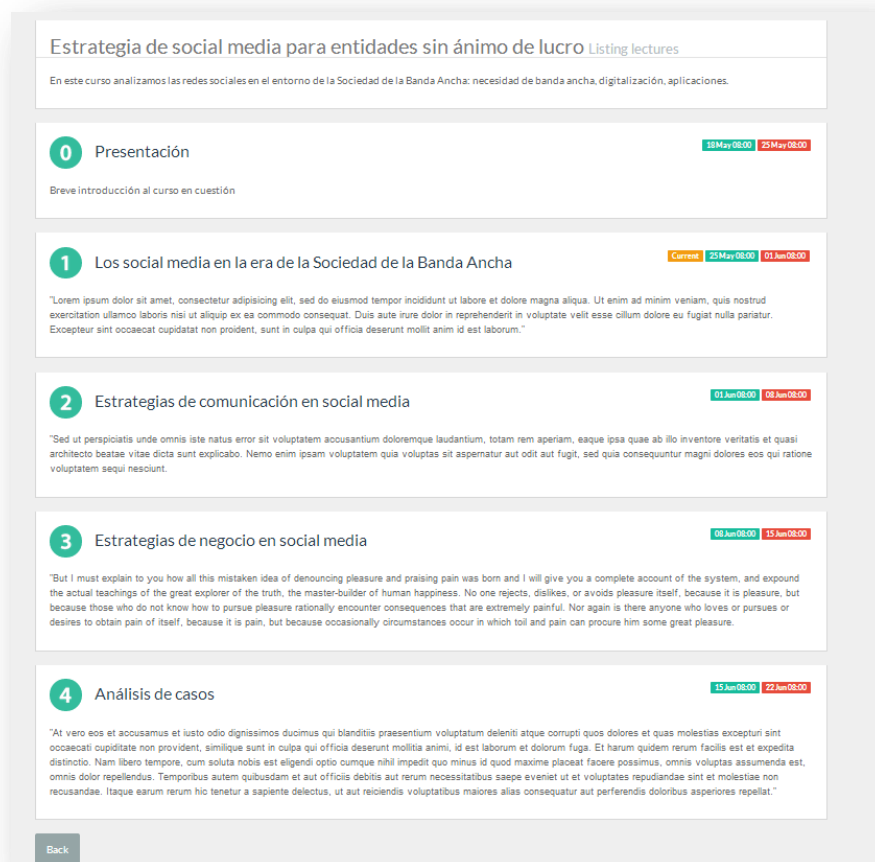


Figura 18

Ver las lecciones

Cuando entramos en un módulo, vemos todo lo preparado por el profesor para poder completar el curso. Solamente veremos lo que el profesor haya creado y hecho visible.

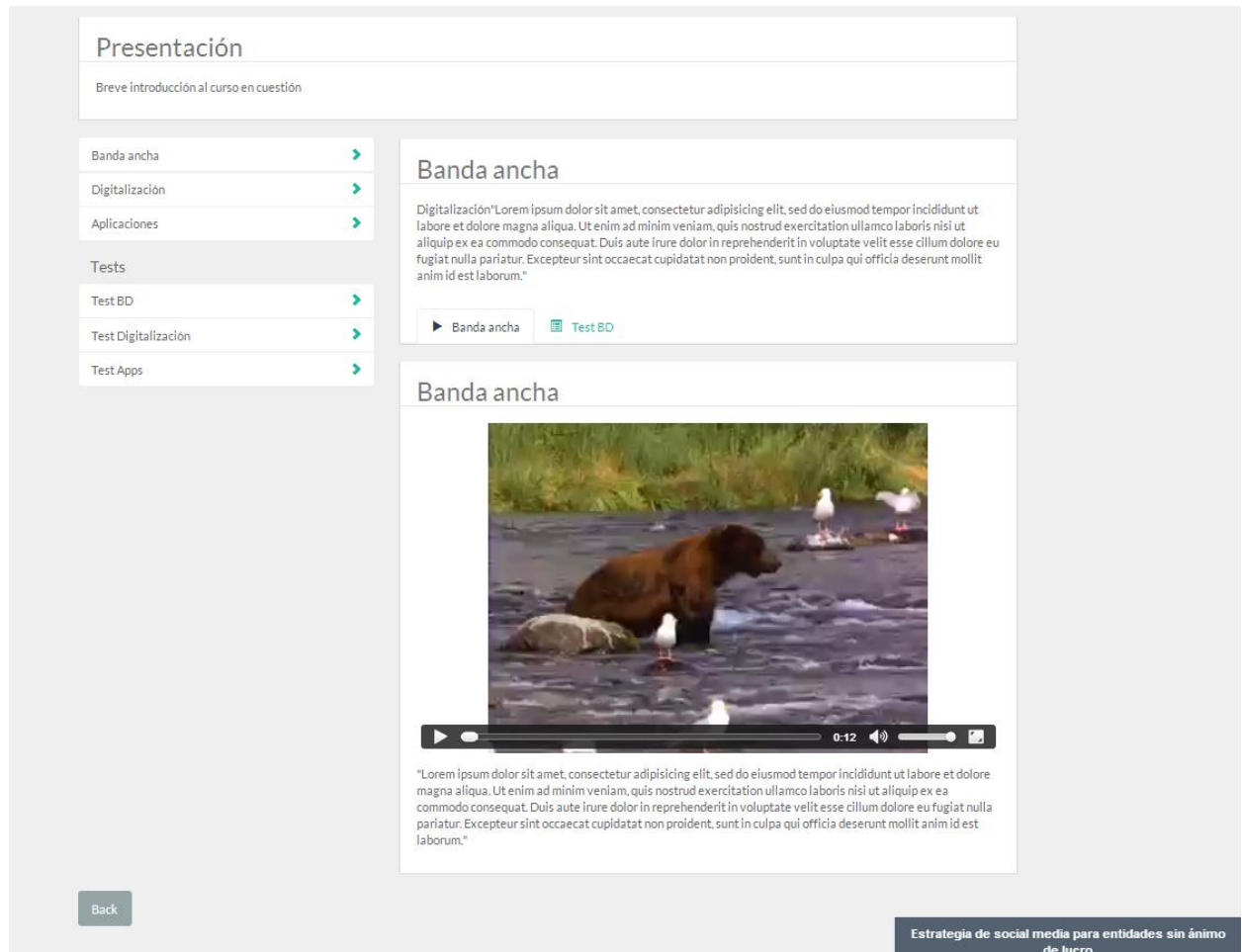


Figura 19

Vemos como se nos despliega a la izquierda una vista con las lecciones y los tests evaluables. Mientras que a la derecha tenemos abierto la sección que hemos seleccionado en el panel de la derecha.

En la ventana superior derecha, tenemos unos seleccionables que nos darán la posibilidad de navegar entre los recursos y los tests de la lección.

Ver los tests

Aquí debemos aclarar dos conceptos clave, por un lado la de los tests que vemos en las lecciones. Y por otro los tests que vemos en el panel de la izquierda, los cuales forman parte del módulo propiamente dicho.

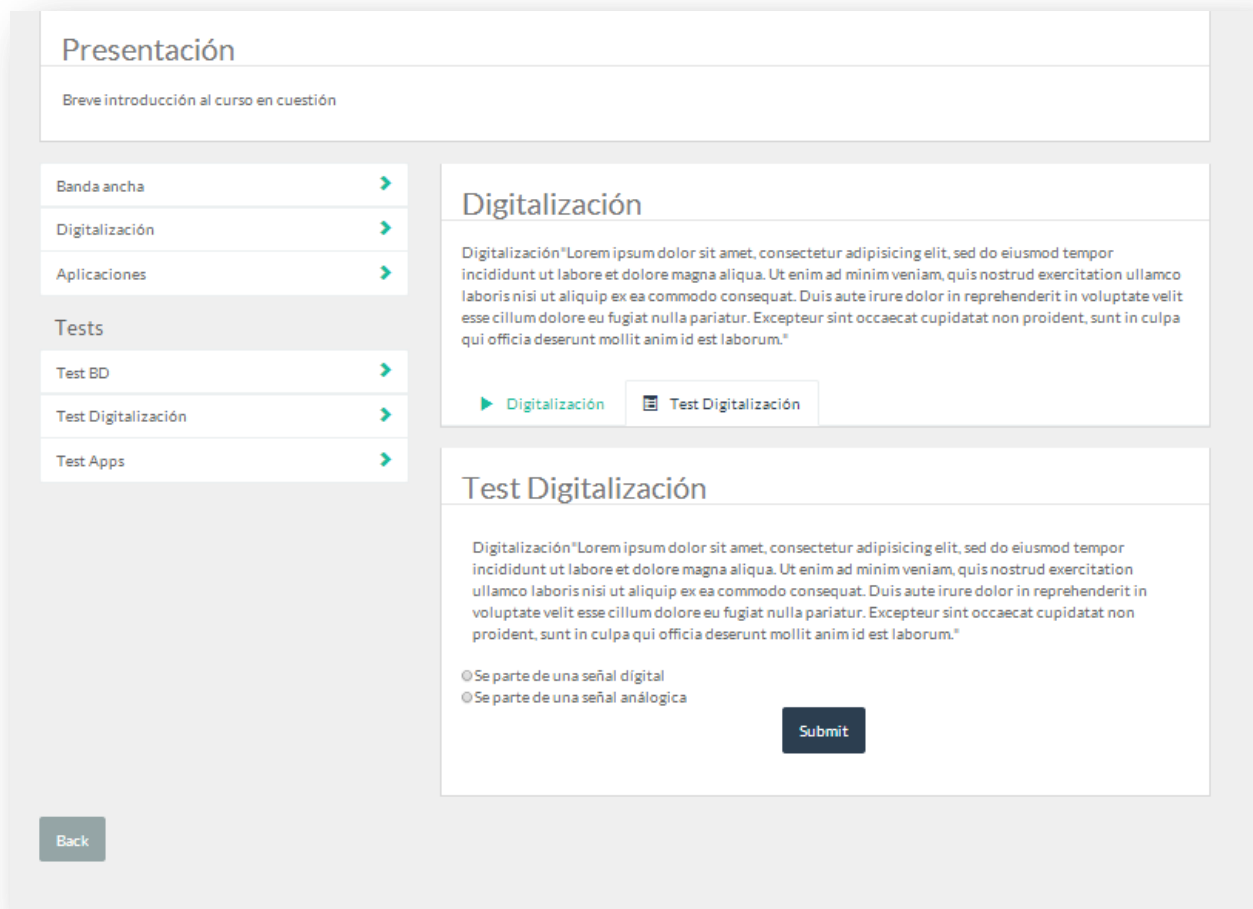
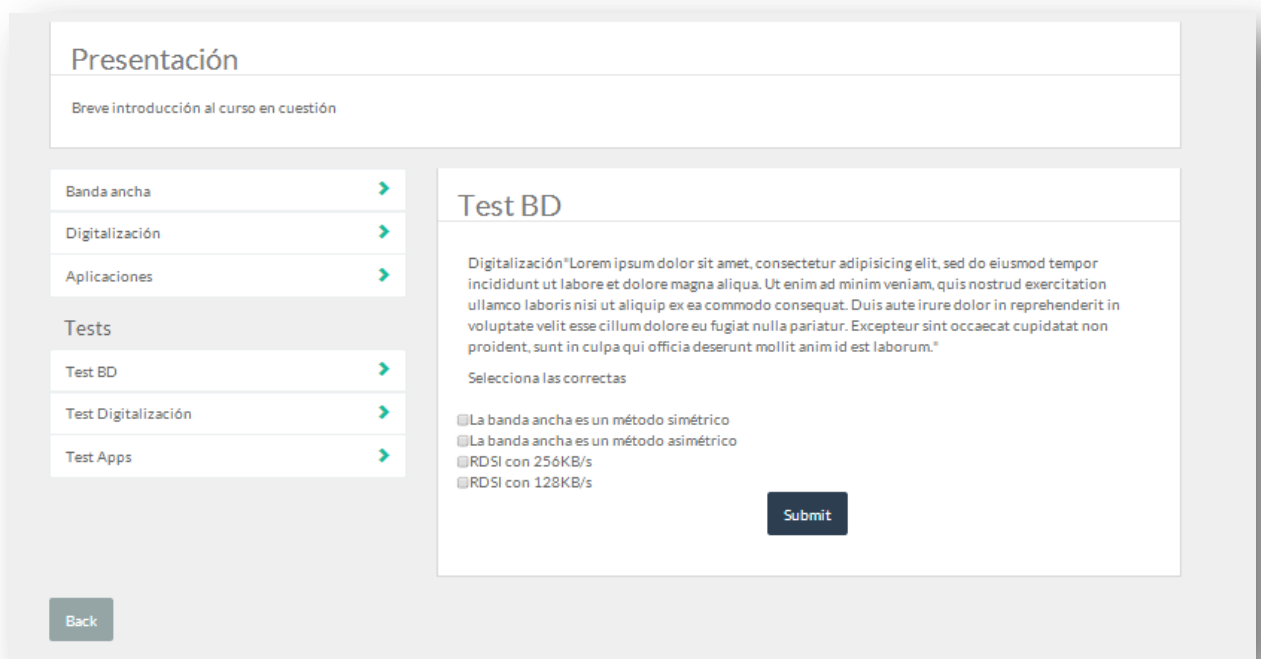


Figura 20

Este tipo de test no es evaluable y puede realizarse tantas veces como se crea necesario.

Por otro lado, tenemos los tests evaluables.

**Figura 21**

Se ve que forma parte del módulo y no de la lección, pues no tenemos el navegador de lecciones.

Estos tests tienen limitaciones de intentos y su resolución forma parte de la nota final del curso.

Durante el curso.

Esta sección va dirigida tanto a profesor como a alumno, y se verán las posibilidades que ofrece el curso una vez empezado.

Noticias

El tablón de noticias será el sustituto de la página principal del curso, siempre y cuando se haya creado alguna noticia del profesor.

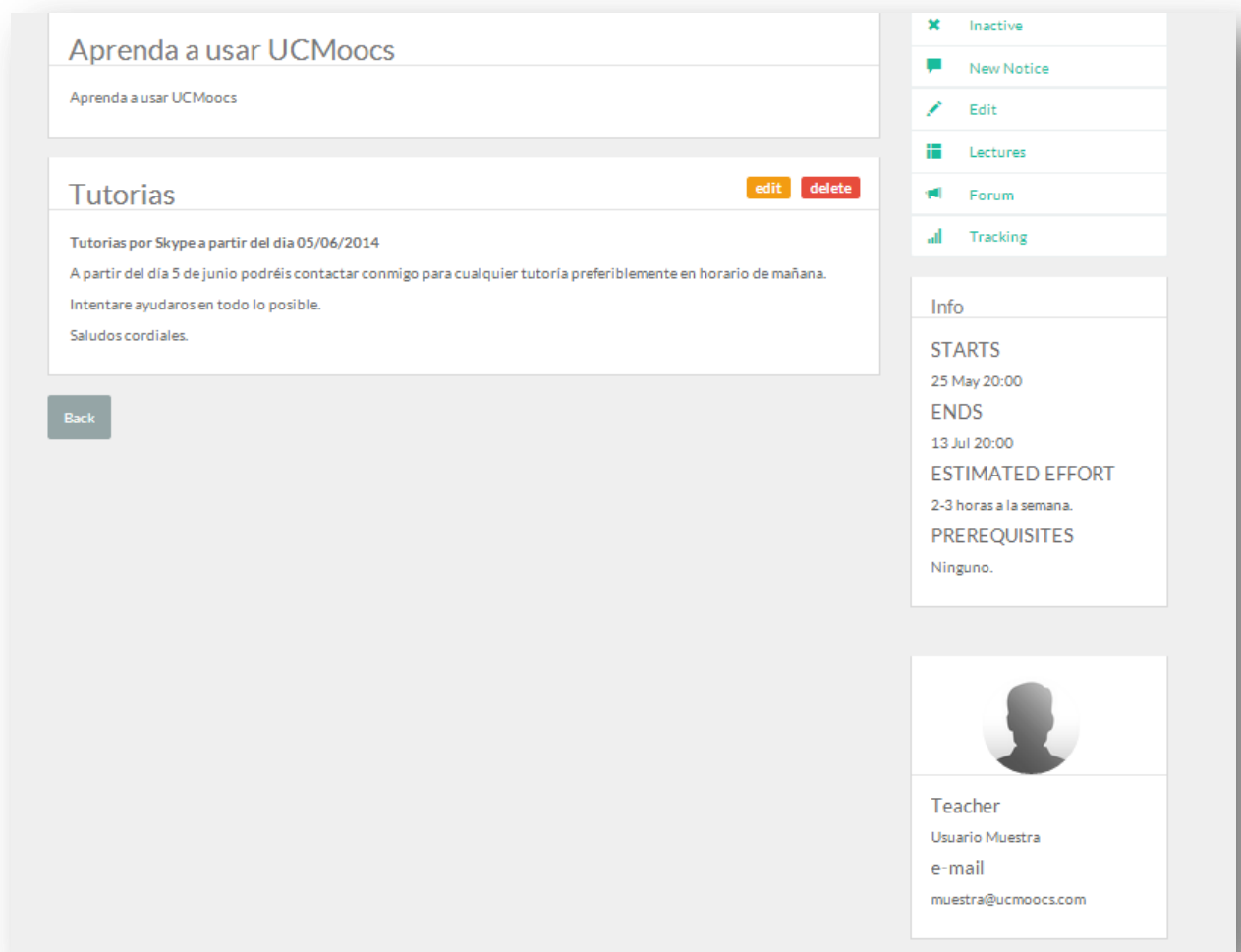


Figura 22

En esta pantalla el profesor podrá comunicarle a los alumnos noticias que crea importantes para la compleción del curso.

Foro.

El foro es parte importante del curso. Un lugar donde poder compartir problemas y/o comentarios adicionales sobre el curso.

Durante la creación del curso el profesor podrá habilitar o deshabilitar el permiso de creación de topics por parte de los usuarios.

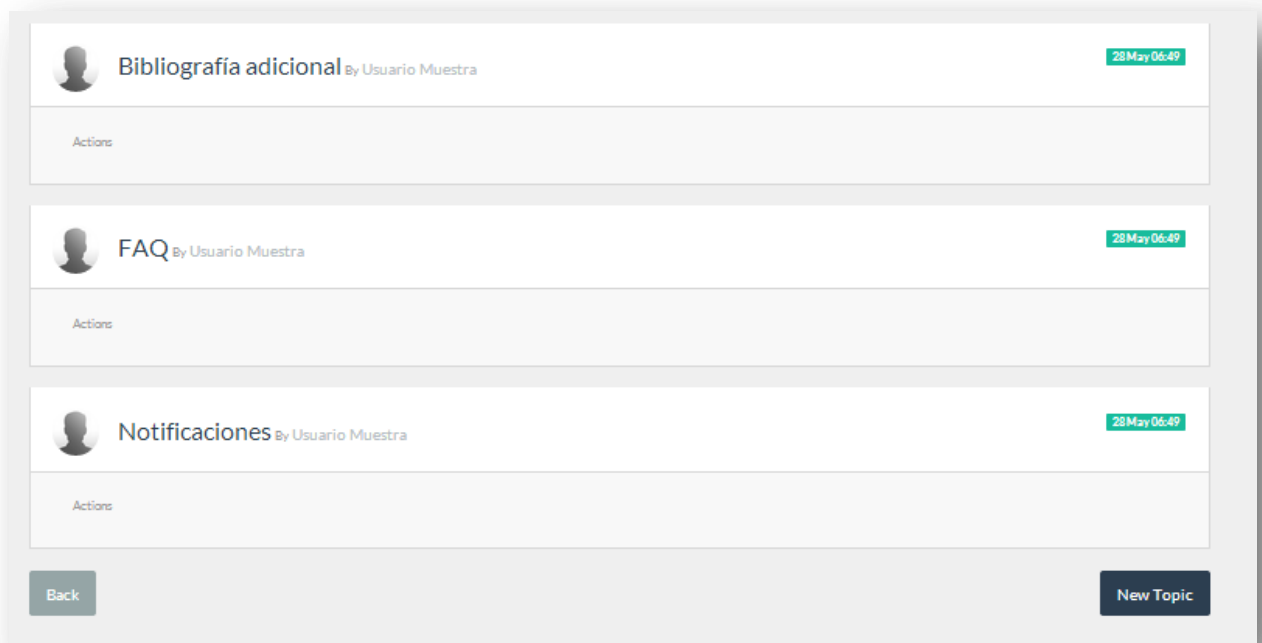


Figura 23

Es importante decir que durante la creación de los topics, hay que habilitar o deshabilitar la posibilidad de que los alumno creen mensajes dentro.

El alumno solo podrá postear donde quiera el profesor.

Evaluación.

La evaluación se hará mediante los tests asociados a módulos.

Tendremos dos tipos de evaluación:

- **Los test de respuesta única.** En estos tests la evaluación se hará dependiendo del número de intentos que se han necesitado para completar el test correctamente.

Test Digitalización

Digitalización"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."

Score: 10.0

☒ Se parte de una señal digital
✓

☐ Se parte de una señal analógica
✗

Figura 24

- **Los test multirespuesta.** En estos tests se tendrán en cuenta las respuestas correctas e incorrectas, restándole al valor de las respuestas correctas, una parte del valor de las respuestas incorrectas.

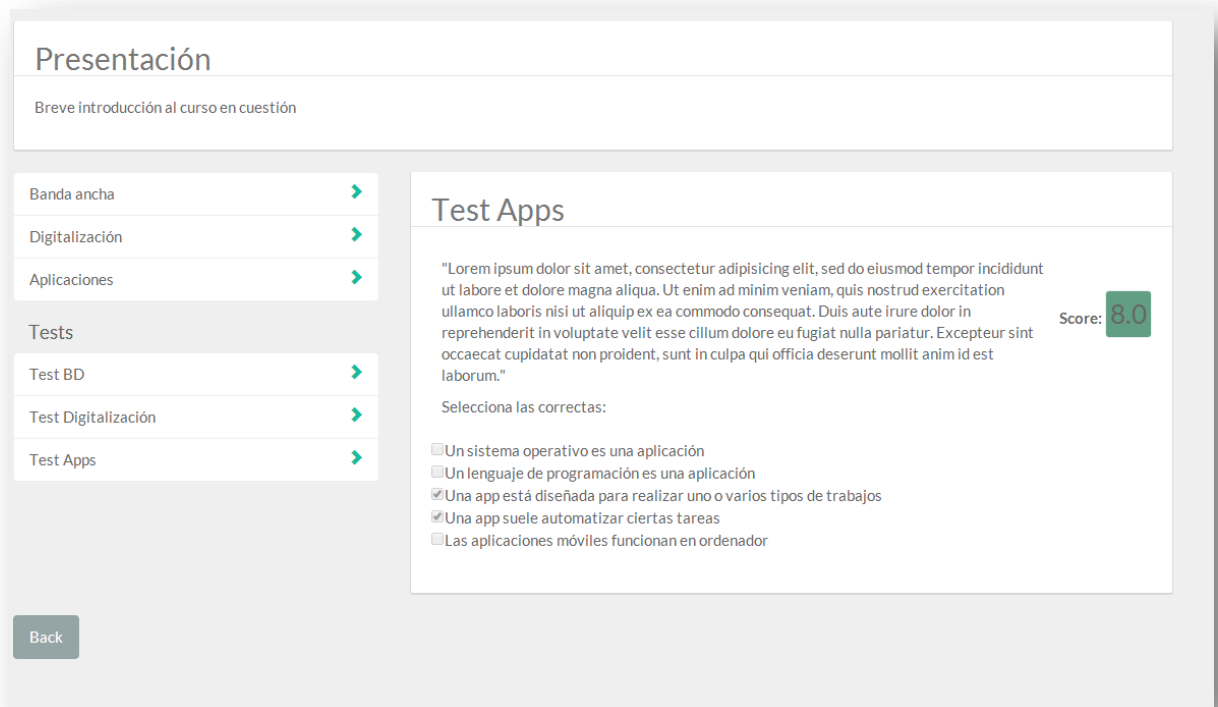


Figura 25

Seguimiento.

En esta pestaña podremos ver el seguimiento del curso.

Si somos un alumno, veremos nuestros resultados por test, así como los mejores, peores y la media de dichos tests. Adicionalmente, tendremos la media de nuestro curso hasta el momento.

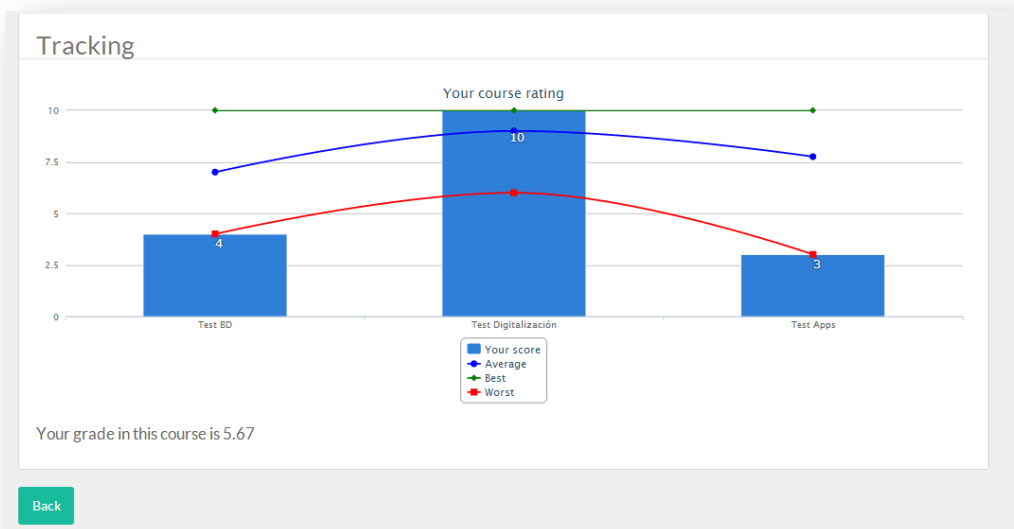


Figura 26

Sin embargo, si somos el profesor, veremos solo los máximos, mínimos y media, y además, por intervalos, la nota media de los alumnos del curso.

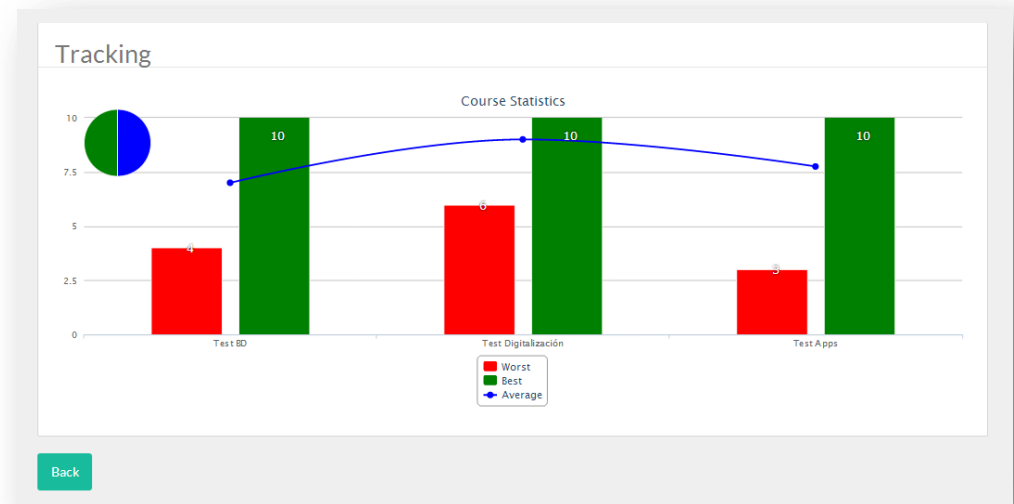


Figura 27

FAQ

¿Hay cosas que aun no entiendo tras haberme leído el manual?

Simplemente puedes enviar un mensaje a través del botón de feedback de la aplicación. Su uso viene explicado en la sección primeras impresiones de este manual.

¿Se bloquean algunas páginas?

Intente recargar la página y reportenos el error para solucionarlo. Si aún persiste vuelva a dirigirse a la página principal.

¿Debo reportar problemas con la información de algún curso?

Lo mejor sería ponerse en contacto con su profesor en el curso, si aun así cree que merece ser reportado, estaremos encantados de recibir sus notificaciones.



Manual de Instalación de la plataforma UCMoocs en un servidor Debian/Ubuntu

Para desplegar la plataforma UCMoocs en un servidor Debian o Ubuntu se requieren diversos componentes que, por regla general, no se encontrarán pre-instalados en el sistema.

La aplicación se encuentra actualmente desplegada para su uso y testing en el siguiente sitio web: <http://moocs.herokuapp.com/>

Para instalar todos los componentes necesarios se han de seguir los siguientes pasos a llevar a cabo en un terminal del sistema:

1. Instalar RVM (Ruby Version Manager). <http://rvm.io/>

```
curl -sSL https://get.rvm.io | bash -s stable
```

2. Instalar la versión 2.0.0 de Ruby

```
rvm install 2.0.0
```

3. Marcar por defecto la versión 2.0.0 de Ruby

```
rvm default 2.0.0
```

4. Instalar Rails

```
gem install rails
```

5. Instalar MongoDB.

<http://docs.mongodb.org/manual/tutorial/install-mongodb-on-ubuntu/>

```
sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 7F0CEB10
```

```
echo 'deb http://downloads-distro.mongodb.org/repo/ubuntu-upstart dist 10gen' | sudo  
tee /etc/apt/sources.list.d/mongodb.list
```

```
sudo apt-get update
```

```
sudo apt-get install mongodb-10gen
```



6. Instalar un intérprete de JavaScript (como nodejs).

sudo apt-get install nodejs

7. Bajar el código fuente de github. <https://github.com/felixmartin/moocs>

git clone <https://github.com/felixmartin/moocs.git>

8. Realizar la instalación de requisitos.

Para esto, situarse en la carpeta “moocs”, que es donde se encuentra el código y ejecutar:

bundle install

9. Seguir los manuales de instalación de Phusion Passenger:

a. Para apache:

<https://www.phusionpassenger.com/documentation/Users%20guide%20Apache.html>

b. Para nginx:

<https://www.phusionpassenger.com/documentation/Users%20guide%20Nginx.html>



Manual de despliegue de la plataforma UCMoocs en Heroku

La guía que se presenta a continuación explica como desplegar la aplicación UCMoocs en un sitio web de Heroku. Se ha realizado basándose en la guía proporcionada por el propio sitio web, la cual se puede consultar en el siguiente enlace:

<https://devcenter.heroku.com/articles/getting-started-with-rails4>

Recordamos que la aplicación se encuentra actualmente desplegada para su uso y testing en el siguiente sitio web de Heroku: <http://moocs.herokuapp.com/>

Para llevar a cabo el despliegue de la aplicación en Heroku se han de seguir los siguientes pasos:

1. **Registrarse en el sitio web:** www.heroku.com
2. **Descargar la herramienta de Heroku para el sistema operativo en el cual se encuentra el código fuente:**
 - Windows: <https://devcenter.heroku.com/toolbelt-downloads/windows>
 - Debian/Ubuntu: <https://devcenter.heroku.com/toolbelt-downloads/debian>
 - OSX: <https://devcenter.heroku.com/toolbelt-downloads/osx>
3. **Descargar un cliente de git:** <http://git-scm.com/>
4. **Desde un terminal, introducir las credenciales de registro del paso 1:**

```
$ heroku login
Enter your Heroku credentials.
Email: user@example.com
Password:
Could not find an existing public key.
Would you like to generate one? [Yn]
Generating new SSH public key.
Uploading ssh public key /Users/user/.ssh/id_rsa.pub
```



5. Inicializar el repositorio:

```
$ git init
$ git add .
$ git commit -m "Deploy"
```

6. Verificar que se ha creado correctamente:

```
$ git status
# On branch master
nothing to commit, working directory clean
```

7. Crear una aplicación en heroku (sustituir la dirección y el repositorio por aquellos que convengan) :

```
heroku create
Creating warm-wave-1943... done, stack is cedar
http://warm-wave-1943.herokuapp.com/ | git@heroku.com:warm-wave-1943.git
Git remote heroku added
```

8. Verificar que el repositorio remoto se ha añadido correctamente.

```
$ git config -e
```

9. Enviar el código de UCMoocs a Heroku (colocarse en la carpeta “master” y sustituir el nombre por el que convenga) :

```
$ git push heroku master
Initializing repository, done.

----> Ruby app detected
----> Compiling Ruby/Rails
----> Using Ruby version: ruby-2.1.1
----> Installing dependencies using 1.5.2
```



```
Running: bundle install --without development:test --path vendor/bundle --
binstubs vendor/bundle/bin -j4 --deployment
Fetching gem metadata from https://rubygems.org/.....
Fetching additional metadata from https://rubygems.org/..
Using json (1.8.1)
Installing i18n (0.6.9)
Installing rake (10.3.0)
Installing builder (3.2.2)
Installing minitest (5.3.3)
Installing thread_safe (0.3.3)
Installing mime-types (1.25.1)
Installing erubis (2.7.0)
Installing polyglot (0.3.4)
Installing arel (5.0.1.20140414130214)
Installing execjs (2.0.2)
Installing coffee-script-source (1.7.0)
Installing rack (1.5.2)
Installing hike (1.2.3)
Using bundler (1.5.2)
Installing thor (0.19.1)
Installing multi_json (1.9.2)
Installing tilt (1.4.1)
Installing rails_serve_static_assets (0.0.2)
Installing rails_stdout_logging (0.0.3)
Installing sass (3.2.19)
Installing rdoc (4.1.1)
Installing tzinfo (1.1.0)
Installing uglifier (2.5.0)
Installing treetop (1.4.15)
Installing coffee-script (2.2.0)
Installing rack-test (0.6.2)
Installing rails_12factor (0.0.2)
Installing sprockets (2.11.0)
Installing sdoc (0.4.0)
Installing activesupport (4.1.0)
Installing mail (2.5.4)
Installing activemodel (4.1.0)
Installing jbuilder (2.0.6)
Installing actionview (4.1.0)
Installing activerecord (4.1.0)
Installing actionpack (4.1.0)
Installing sprockets-rails (2.1.3)
Installing actionmailer (4.1.0)
Installing railties (4.1.0)
Installing coffee-rails (4.0.1)
Installing jquery-rails (3.1.0)
Installing turbolinks (2.2.2)
Installing sass-rails (4.0.3)
Installing rails (4.1.0)
```



```
Installing pg (0.17.1)
Your bundle is complete!
Gems in the groups development and test were not installed.
It was installed into ./vendor/bundle
Post-install message from rdoc:
Depending on your version of ruby, you may need to install ruby rdoc/ri
data:
<= 1.8.6 : unsupported
= 1.8.7 : gem install rdoc-data; rdoc-data --install
= 1.9.1 : gem install rdoc-data; rdoc-data --install
>= 1.9.2 : nothing to do! Yay!
Bundle completed (48.17s)
Cleaning up the bundler cache.
-----> Preparing app for Rails asset pipeline
Running: rake assets:precompile
I, [2014-04-17T17:54:07.273555 #1263] INFO -- : Writing
/tmp/build_8945fb34-0238-41d7-a2fd-4402581a95fe/public/assets/application-
10faafa06109fa14582542ac1852f5c5.js
I, [2014-04-17T17:54:07.338617 #1263] INFO -- : Writing
/tmp/build_8945fb34-0238-41d7-a2fd-4402581a95fe/public/assets/application-
9cc0575249625b8d8648563841072913.css
Asset precompilation completed (11.26s)
Cleaning assets
Running: rake assets:clean
-----> WARNINGS:
No Procfile detected, using the default web server (webrick)
https://devcenter.heroku.com/articles/ruby-default-web-server
-----> Discovering process types
Procfile declares types -> (none)
Default types for Ruby -> console, rake, web, worker

-----> Compressing... done, 21.9MB
-----> Launching... done, v6
http://warm-wave-1943.herokuapp.com/ deployed to Heroku

To git@heroku.com:warm-wave-1943.git
* [new branch]      master -> master
```

10. Desde el panel de control de Heroku, configurar MongoLAB para que esté activo y visitar la aplicación desplegada.